

5-6 设计消息模块的Web层

140.143.132.225:8000/project-1/doc-91

一、获取分页消息列表

创建 `SearchMessageByPageForm.java` 类

```
1. @ApiModelProperty
2. @Data
3. public class SearchMessageByPageForm {
4.     @NotNull
5.     @Min(1)
6.     private Integer page;
7.     @NotNull
8.     @Range(min = 1,max = 40)
9.     private Integer length;
10. }
```

创建 `MessageController.java` 类

```
1. @RestController
2. @RequestMapping("/message")
3. @Api("消息模块网络接口")
4. public class MessageController {
5.     @Autowired
6.     private JwtUtil jwtUtil;
7.     @Autowired
8.     private MessageService messageService;
9.     @PostMapping("/searchMessageByPage")
10.    @ApiOperation("获取分页消息列表")
11.    public R searchMessageByPage(@Valid @RequestBody SearchMessageByPageForm form,
12.    @RequestHeader("token") String token) {
13.        int userId = jwtUtil.getUserId(token);
14.        int page = form.getPage();
15.        int length = form.getLength();
16.        long start = (page - 1) * length;
17.        List<HashMap> list = messageService.searchMessageByPage(userId, start, length);
18.        return R.ok().put("result", list);
19.    }
```

二、根据ID查询消息

创建 `SearchMessageByIdForm.java` 类

```
1. @ApiModelProperty
2. @Data
3. public class SearchMessageByIdForm {
4.     @NotBlank
5.     private String id;
6. }
```

在 `MessageController.java` 中编写Web方法

```
1. public class MessageController {
2.     .....
3.     @PostMapping("/searchMessageById")
4.     @ApiOperation("根据ID查询消息")
5.     public R searchMessageById(@Valid @RequestBody SearchMessageByIdForm form) {
6.         HashMap map = messageService.searchMessageById(form.getId());
7.         return R.ok().put("result", map);
8.     }
9. }
```

三、把未读消息更新成已读消息

创建 `UpdateUnreadMessageForm.java` 类

```
1. @ApiModelProperty
2. @Data
3. public class UpdateUnreadMessageForm {
4.     @NotBlank
5.     private String id;
6. }
```

在 `MessageController.java` 中编写Web方法

```
1. public class MessageController {
2.     .....
3.     @PostMapping("/updateUnreadMessage")
4.     @ApiOperation("未读消息更新成已读消息")
5.     public R updateUnreadMessage(@Valid @RequestBody UpdateUnreadMessageForm form) {
6.         long rows = messageService.updateUnreadMessage(form.getId());
7.         return R.ok().put("result", rows == 1 ? true : false);
8.     }
9. }
```

四、删除消息

创建 `DeleteMessageRefByIdForm.java` 类

```
1. @Data
2. @ApiModel
3. public class DeleteMessageRefByIdForm {
4.     @NotBlank
5.     private String id;
6. }
```

在 `MessageController.java` 中编写Web方法

```
1. public class MessageController {
2.     .....
3.     @PostMapping("/deleteMessageRefById")
4.     @ApiOperation("删除消息")
5.     public R deleteMessageRefById(@Valid @RequestBody DeleteMessageRefByIdForm form){
6.         long rows=messageService.deleteMessageRefById(form.getId());
7.         return R.ok().put("result", rows == 1 ? true : false);
8.     }
9. }
```

