以猫为主题的富媒体分享社区

测试用例

版本 <1.0>

修订历史记录

|  |  |  |  |
| --- | --- | --- | --- |
| **日期** | **版本** | **说明** | **作者** |
| <09/08/17> | <1.0> |  | 全体成员 |
| <09/10/17> | <2.0> |  | 俞昕宜 |
|  |  |  |  |
|  |  |  |  |

# 简介

## 目的

通过测试来验证开发系统的正确性、完整性、安全性。

## 范围

测试项目包括“以猫为主题的富媒体社交平台”系统的后端系统与iOS客户端。

## 定义、首字母缩写词和缩略语

点滴：以图或视频为主的用户分享，可包含少量文字描述

文章：以文字为主的用户分享，可在段落中插入图片

问答：一个用户提出的问题及其相应的其他用户回答

问题：由用户提出，以问号结尾，可带有描述性文字的问题，提出后将由所有用户主动解答

回答：用户针对其他用户提出的问题而作出的回答，以文字为主，段落中可插入图片以辅助说明

收藏：用户收藏文章和回答后可在“我的收藏”页面快速访问

## 概述

测试用例分为“iOS客户端单元测试”、“iOS客户端集成测试”与“后端单元测试”，本文档将按照此分类进行组织。

# iOS客户端单元测试

SearchHistorySource

名称：testAddHistory

测试输入：addHistory(historyEntry:)

预期结果：搜索历史中包含搜索关键字

名称：testClearHistory

测试输入：clearHistory()

预期结果：搜索历史为空

UserDefaultExtesnsion

名称: testCustomObject

测试输入: setCustomObject(forKey:)

预期结果：customObject(forKey:) 返回输入对象

QiniuProvider

名称：testQiniuProvider

测试输入：upload(data:)

预期结果：返回有效的Observable对象，不抛出异常

Token

名称: testToken

测试输入: Token.save

预期结果：Token.load 正确加载保存的 token

Logger

名称：testDateFormatter

测试输入：当前时间

预期结果：按 “yyyy-MM-dd HH:mm:ss.SSS” 格式化字符串

名称: testLogger

测试输入: log(message:)

预期结果：最后一条logger在log文件尾部

名称：testLogPath

测试输入：logPath

预期结果：返回合法文件路径

MeowAPIProvider

名称：testRefresh

测试输入：refresh

预期结果：单例中token 与 Token.load 一致

名称：testRequest

测试输入：request

预期结果：返回Endpoint，包含正确的 Http Method、url、request body

JSONConvertible

名称：testInt

测试输入：JSON数据

预期结果：合法输入返回 Parse 后的 Int 类型对象，非法输入返回 nil

名称：testFloat

测试输入：JSON数据

预期结果：合法输入返回 Parse 后的 Float 类型对象，非法输入返回 nil

名称：testDouble

测试输入：JSON数据

预期结果：合法输入返回 Parse 后的 Double 类型对象，非法输入返回 nil

名称：testBool

测试输入：JSON数据

预期结果：合法输入返回 Parse 后的 Bool 类型对象，非法输入返回 nil

名称：testString

测试输入：JSON数据

预期结果：合法输入返回 Parse 后的 String 类型对象，非法输入返回 nil

名称：testUrl

测试输入：JSON数据

预期结果：合法输入返回 Parse 后的 Url 类型对象，非法输入返回 nil

名称：testDate

测试输入：JSON数据

预期结果：合法输入返回 Parse 后的 Date 类型对象，非法输入返回 nil

名称：testAnswer

测试输入：JSON数据

预期结果：合法输入返回 Parse 后的 Answer 类型对象，非法输入返回 nil

名称：testMoment

测试输入：JSON数据

预期结果：合法输入返回 Parse 后的 Moment 类型对象，非法输入返回 nil

名称：testArticle

测试输入：JSON数据

预期结果：合法输入返回 Parse 后的 Article 类型对象，非法输入返回 nil

名称：testQuestion

测试输入：JSON数据

预期结果：合法输入返回 Parse 后的 Question 类型对象，非法输入返回 nil

名称：testProfile

测试输入：JSON数据

预期结果：合法输入返回 Parse 后的 Profile 类型对象，非法输入返回 nil

名称：testMedia

测试输入：JSON数据

预期结果：合法输入返回 Parse 后的 Media 类型对象，非法输入返回 nil

名称：testBanner

测试输入：JSON数据

预期结果：合法输入返回 Parse 后的 Banner 类型对象，非法输入返回 nil

名称：testUploadToken 测试输入：JSON数据

预期结果：合法输入返回 Parse 后的 UploadToken 类型对象，非法输入返回 nil

名称：testComment

测试输入：JSON数据

预期结果：合法输入返回 Parse 后的 Comment 类型对象，非法输入返回 nil

名称：testAnswerSummary

测试输入：JSON数据

预期结果：合法输入返回 Parse 后的 AnswerSummary 类型对象，非法输入返回 nil

名称：testArticleSummary

测试输入：JSON数据

预期结果：合法输入返回 Parse 后的 ArticleSummary 类型对象，非法输入返回 nil

ChatKitUser

名称：testChatKitUser

测试输入：init(profile:)

预期结果：userId, nickname, avatar 属性与 profile 一致

# iOS客户端集成测试

* 1. *<登陆 测试用例>*

名称：登陆测试

前置条件：用户未登录  
测试输入：输入用户名、密码

观察点：点击“登陆”

控制点：输入信息是否正确

预期结果：若*正*确，进入首页；若错误，显示错误提示

* 1. *<注册 测试用例>*

名称：注册测试  
前置条件：用户未登录  
测试输入：输入无效的手机号、验证码或密码  
观察点：点击“注册”  
预期结果：显示错误提示

* 1. *<浏览首页点滴 测试用例>*

名称：浏览首页点滴  
前置条件：用户已登陆  
测试输入：点击底栏“首页”，上下滑动  
预期结果：显示保留在页面顶端的搜索栏及发布按钮；显示可滑动的封面页；显示多个点滴文字及合理排布的图片；显示发布用户基本信息和发布时间；显示点赞数与评论数；显示评论框

* 1. *<点滴详情 测试用例>*

名称：点滴详情测试  
前置条件：用户浏览首页  
测试输入：点击一条点滴  
预期结果：显示点滴详情（与首页几近相同），包含用户评论

* 1. *<浏览文章 测试用例>*

名称：浏览文章测试  
前置条件：用户已登陆  
测试输入：点击底栏“文章”，上下滑动  
预期结果：显示保留在页面顶端的发布按钮；显示合理排布的多篇文章封面图片、标题、简介、点赞数、评论数

* 1. *<文章详情 测试用例>*

名称：文章详情测试  
前置条件：用户浏览文章页  
测试输入：点击一篇文章  
预期结果：显示发表文章用户信息及发布时间；显示文字图片合理排布的文章内容；显示赞、评论、收藏、更多等按钮

* 1. *<浏览问答 测试用例>*

名称：浏览问答测试  
前置条件：用户已登陆  
测试输入：点击底栏“问答”，上下滑动  
预期结果：显示保留在页面顶端的发布按钮；显示合理排布的多个问答标题、回答；显示点赞数、评论数

* 1. *<问答详情 测试用例>*

名称：问答详情测试前置条件：用户浏览问答页  
测试输入：点击一个问答  
预期结果：显示问答详情

* 1. *<回答详情 测试用例>*

名称：回答详情测试  
前置条件：用户在某一问答详情页  
测试输入：点击一个回答  
预期结果：显示发表回答用户信息及发布时间；显示文字图片合理排布的回答内容；显示赞、评论、收藏、更多等按钮

* 1. *<点赞 测试用例>*

名称：点赞测试  
前置条件：用户在首页／用户在点滴详情页／用户在文章详情页／用户在回答详情页／用户在其他用户个人信息点滴页／用户在我的点滴页

测试输入：点击“点赞”按钮  
预期结果：“赞”按钮变为“已赞”；显示点赞数增加一；切换页面／重新开启后赞数、点赞按钮显示正确  
测试输入：点击“已赞”按钮  
预期结果：“已赞”按钮变为“赞”；显示点赞数减少一；切换页面／重新开启后赞数、点赞按钮显示正确

* 1. *<评论 测试用例>*
     1. 名称：评论测试  
        前置条件：用户在首页／用户在我的点滴页  
        测试输入：用户输入评论，按下回车  
        预期结果：显示评论数增加一；评论框清空；进入详情页可以查看到该评论
     2. 名称：详情评论测试  
        前置条件：用户在点滴详情页  
        测试输入：用户输入评论，按下回车  
        预期结果：显示评论数增加一；评论框清空；该页显示该评论
     3. 名称：评论测试  
        前置条件：用户在文章详情页／用户在回答详情页  
        测试输入：用户点击评论  
        预期结果：正确显示评论页面  
        测试输入：用户点击加号，输入评论，点击完成  
        预期结果：输入页面关闭；评论页面正确显示该评论
  2. *<收藏 测试用例>*

名称：收藏测试  
前置条件：用户在文章详情页／用户在回答详情页  
测试输入：点击“收藏”按钮  
预期结果：“收藏”按钮变为“已收藏”；显示收藏数增加一；切换页面／重新开启后收藏数、收藏按钮显示正确  
测试输入：点击“已收藏”按钮  
预期结果：“已收藏”按钮变为“收藏”；显示收藏数减少一；切换页面／重新开启收藏数、收藏按钮显示正确

* 1. *<举报 测试用例>*

名称：举报测试  
前置条件：用户在文章详情页／用户在回答详情页  
测试输入：点击“举报”，输入举报理由，点击“好”  
预期结果：举报输入框消失；管理员收到该举报

* 1. *<分享 测试用例>*

名称：分享测试  
前置条件：用户在文章详情页／用户在回答详情页  
测试输入：点击右下方“More”，点击分享  
预期结果：出现分享对话框，对话框内显示分享提示信息

* 1. *<关注问答 测试用例>*

名称：问答详情页关注测试  
前置条件：用户在问答详情页  
测试输入：多次点击“关注”／“取消关注”

预期结果：按钮文字由“关注”变为“取消关注”， 或由“取消关注”变为“关注”；用户可在个人页“我的关注”中查看到相应变化

* 1. *<发布页 测试用例>*

名称：发布页测试  
前置条件：用户已登陆  
测试输入：用户点击右上角的“+”  
预期结果：显示发布选择页面，可选择发布点滴、文章、问题

* 1. *<发布点滴 测试用例>*

名称：发布点滴测试  
前置条件：用户进入发布页  
测试输入：用户输入点滴内容，选择图片，点击“发布”  
预期结果：输入页面关闭；用户可在首页查看到该点滴；用户可在个人页“我的点滴”查看到该点滴；该点滴文字图片等信息显示正确

* 1. *<发布文章 测试用例>*

名称：发布文章测试  
前置条件：用户进入发布页  
测试输入：用户输入文章标题内容，选择图片，点击“发布”  
预期结果：输入页面关闭；用户可在文章页查看到该文章；用户可在个人页“我的文章”查看到该文章；该文章文字图片等信息显示正确「封面图？」

* 1. *<发布问题 测试用例>*

名称：发布问题测试  
前置条件：用户已登陆，用户进入发布页  
测试输入：用户输入问题标题内容，点击“发布”  
预期结果：输入页面关闭；用户可在个人页“我的问答”“我的问题”中查看到该问题；该问题显示正确

* 1. *<参与问答 测试用例>*

名称：参与问答测试  
前置条件：用户进入问题详情页*>*测试输入：用户点击“回答”，输入回答，点击“完成”  
预期结果：显示输入页面；输入页面关闭；在该问题详情页可查看到该回答；在“我的问答”“我的回答”中可查看到该回答

* 1. *<搜索结果页 测试用例>*

名称：搜索结果页点击测试   
前置条件：用户已登陆  
测试输入：用户输入搜索内容，进行搜索  
预期结果：搜索页正确显示多种搜索结果，可能包含点滴文章及问答；每个搜索结果均可点开查看详情

* 1. *<搜索算法 测试用例>*

名称：搜索算法测试   
前置条件：用户已登陆  
测试输入：用户输入搜索内容，进行搜索  
预期结果：点开每个搜索结果，均包含相关搜索字符；  
测试输入：用户新发表某一内容，并搜索该内容相关字符  
预期结果：搜索结果显示该项目

* 1. *<搜索历史记录 测试用例>*

名称：搜索历史记录测试  
前置条件：用户已登陆，用户在搜索页面  
测试输入：用户输入搜索内容，搜索后返回搜索页面  
预期结果：历史记录包含该搜索字符  
测试输入：用户点击某一搜索记录  
预期结果：搜索框内显示该字符  
测试输入：用户点击“清空历史记录”  
预期结果：历史记录已清空

* 1. *<其他用户个人信息 测试用例>*

名称：其他用户个人信息测试  
前置条件：用户在首页／用户在点滴详情页／用户在文章详情页／用户在回答详情页  
测试输入：用户点击其他用户头像  
预期结果：显示该用户的个人主页，包括基本资料及“她的动态”“她的文章”“她的回答”

* 1. *<浏览用户动态 测试用例>*

名称：浏览用户动态测试  
前置条件：用户进入其他用户个人主页  
测试输入：用户点击“她的动态”“她的文章”“她的回答”，上下滑动  
预期结果：页面均能正确合理显示相应信息；可以点开相应条目

* 1. *<关注用户 测试用例>*

名称：关注用户测试  
前置条件：用户进入其他用户个人主页  
测试输入：多次点击“关注”／“取消关注”

预期结果：按钮文字由“关注”变为“取消关注”， 或由“取消关注”变为“关注”；用户可在个人页“我的关注”中查看到相应变化

* 1. *<私信用户 测试用例>*

名称：私信用户测试  
前置条件：用户进入其他用户个人主页  
测试输入：用户点击“私信”  
预期结果：出现聊天页面  
测试输入：用户输入聊天内容，点击发送  
预期结果：消息发送成功；对方用户可以查看

* 1. *<查看个人信息 测试用例>*

名称：查看个人信息测试

前置条件：用户已登陆  
测试输入：点击底栏“我的”按钮  
预期结果：正确显示我的个人信息

* 1. *<管理个人信息 测试用例>*

名称：管理个人信息测试  
前置条件：用户在“我的”页面  
测试输入：用户点击个人信息  
预期结果：弹出编辑个人资料窗口，可修改昵称、修改简介  
测试输入：用户分别点击“修改昵称”“修改简介”，输入修改信息提交  
预期结果：个人信息更新，显示正确

* 1. *<浏览个人记录 测试用例>*

名称：浏览个人记录测试  
前置条件：用户在“我的”页面  
测试输入：用户点击“我的点滴”／“我的文章”／“我的问答”  
预期结果：显示用户发布的内容；显示正确；可点击进入详情页

* 1. *<查看个人关注 测试用例>*

名称：查看个人关注测试  
前置条件：用户进入“我的”页面

测试输入：用户点击“我的关注”，分别点击“关注的用户”和“关注的问题”

预期结果：页面显示关注的用户列表或关注的问题列表

* 1. *<查看个人收藏 测试用例>*

名称：查看个人收藏测试  
前置条件：用户进入“我的”页面  
测试输入：用户点击“我的收藏”  
预期结果：页面显示收藏的回答、文章列表

# 后端单元测试

## 测试用例 1

测试StringUtil类的replaceNull方法。

1) 测试输入

i) “str”

ii) null

2) 预期结果

i) “str”

ii) “”

## 测试用例 2

测试StringUtil类的filterRichText方法。

1) 测试输入

i) “<script>alert('hello')</script>”

ii) "<a href=\"[http://example.com\](about:blank)">text</a>"

iii) “<p>blocks</p>”

iv) “<b>formatting</b>”

v) “<img src=\"http://lorempixel.com/50/50\" />”

vi) "<div style=\"color: #3c3;\">styled div</div>"

2) 预期结果

i) “”

ii) "text"

iii) "<p>blocks</p>"

iv) "<b>formatting</b>"

v) "<img src=\"http://lorempixel.com/50/50\" />"

vi) "<div style=\"color:#3c3\">styled div</div>"

## 测试用例 3

测试StringUtil类的extractHTMLSummary方法。

1) 测试输入

i) “<p>paragraph 1</p><p>paragraph 2</p>”

ii) "<p>malformed data"

2) 预期结果

i) “paragraph 1”

ii) ""

## 测试用例 4

测试StringUtil类的wrapLikeSubstr方法。

1) 测试输入

i) "str"

ii) "\_99%"

2) 预期结果

i) "%str%"

ii) "%\\\_99\\%%"

## 测试用例 5

测试StringUtil类的tryStringToPosLong方法。

1) 测试输入

i) "123"

ii) "abc"

2) 预期结果

i) 123

ii) 0

## 测试用例 6

测试StringUtil类的parseReportStatus方法。

1) 测试输入

i) “open”

ii) "closed"

iii) "invalid status"

2) 预期结果

i) false

ii) true

iii) null

## 测试用例 7

测试FormatValidator类的checkNonNegativeInt方法。

1) 测试输入

i) null

ii) 100

iii) 0

iv) -33

2) 预期结果

i) false

ii) true

iii) true

iv) false

## 测试用例 8

测试FormatValidator类的checkPositiveInt方法。

1) 测试输入

i) null

ii) 100

iii) 0

iv) -33

2) 预期结果

i) false

ii) true

iii) false

iv) false

## 测试用例 8

测试FormatValidator类的checkPhone方法。

1) 测试输入

i) null

ii) “123”

iii) "123xxxxyyyy"

iv) "12333333333"

2) 预期结果

i) false

ii) false

iii) false

iv) true

## 测试用例 8

测试FormatValidator类的checkPassword方法。

1) 测试输入

i) null

ii) "password"

iii) "666666"

iv) "123"

2) 预期结果

i) false

ii) true

iii) true

iv) false

## 测试用例 8

测试FormatValidator类的checkSmsCode方法。

1) 测试输入

i) null

ii) "hacode"

iii) "666666"

2) 预期结果

i) false

ii) false

iii) true

## 测试用例 8

测试FormatValidator类的checkMainItemType方法。

1) 测试输入

i) null

ii) -1

iii) 0

iv) 1

v) 2

vi) 3

vii) 4

2) 预期结果

i) false

ii) false

iii) true

iv) true

v) true

vi) true

vii) false

## 测试用例 8

测试FormatValidator类的checkItemType方法。

1) 测试输入

i) null

ii) -1

iii) 0

iv) 1

v) 2

vi) 3

vii) 4

viii) 5

2) 预期结果

i) false

ii) false

iii) true

iv) true

v) true

vi) true

vii) true

viii) false

# 后端集成测试

## 测试用例 1

测试/api/answers接口get方法。

1) 前置条件

登陆。

2) 测试输入

对/api/answers发送get请求，包含page，size参数。

3) 预期结果

收到以给定size进行分页的第page页回答数据。

## 测试用例 2

测试/api/answers/{id}接口get方法。

1) 前置条件

登陆。

2) 测试输入

对/api/answers/{id}发送get请求。

3) 预期结果

收到id为指定参数的回答数据。

## 测试用例 3

测试/api/question/{id}/answers接口post方法。

1) 前置条件

登陆。

2) 测试输入

对/api/question/{id}/answers发送post请求，包含content参数。

3) 预期结果

数据库记录以content为内容的指定id问题的新回答。

## 测试用例 4

测试/api/answers/{id}接口delete方法。

1) 前置条件

登陆。

2) 测试输入

对/api/answers/{id}发送delete请求。

3) 预期结果

数据库删除指定id的回答。

## 测试用例 5

测试/api/answers/{id}/like接口get方法。

1) 前置条件

登陆。

2) 测试输入

对/api/answers/{id}/like发送get请求。

3) 预期结果

返回指定id回答的当前用户是否点赞。

## 测试用例 6

测试/api/answers/{id}/like接口post方法。

1) 前置条件

登陆。

2) 测试输入

对/api/answers/{id}/like发送post请求。

3) 预期结果

数据库记录当前用户点赞指定id回答。

## 测试用例 7

测试/api/answers/{id}/like接口delete方法。

1) 前置条件

登陆。

2) 测试输入

对/api/answers/{id}/like发送delete请求。

3) 预期结果

数据库删除当前用户点赞指定id回答。

## 测试用例 8

测试/api/answers/{id}/favorite接口get方法。

1) 前置条件

登陆。

2) 测试输入

对/api/answers/{id}/favorite发送get请求。

3) 预期结果

返回指定id回答的当前用户是否收藏。

## 测试用例 9

测试/api/answers/{id}/favorite接口post方法。

1) 前置条件

登陆。

2) 测试输入

对/api/answers/{id}/favorite发送post请求。

3) 预期结果

数据库记录当前用户收藏指定id回答。

## 测试用例 10

测试/api/answers/{id}/favorite接口delete方法。

1) 前置条件

登陆。

2) 测试输入

对/api/answers/{id}/favorite发送delete请求。

3) 预期结果

数据库删除当前用户收藏指定id回答。

## 测试用例 11

测试/api/answers/{id}/report接口post方法。

1) 前置条件

登陆。

2) 测试输入

对/api/answers/{id}/report发送post请求，包含itemId、itemType、reason参数。

3) 预期结果

数据库记录该用户对该回答的举报。

## 测试用例 12

测试/api/questions接口get方法。

1) 前置条件

登陆。

2) 测试输入

对/api/questions发送get请求，包含page，size参数。

3) 预期结果

收到以给定size进行分页的第page页问题数据。

## 测试用例 13

测试/api/questions/{id}接口get方法。

1) 前置条件

登陆。

2) 测试输入

对/api/questions/{id}发送get请求。

3) 预期结果

收到id为指定参数的问题数据。

## 测试用例 14

测试/api/question接口post方法。

1) 前置条件

登陆。

2) 测试输入

对/api/question发送post请求，包含title、content参数。

3) 预期结果

数据库记录以title为标题、以content为内容的新问题。

## 测试用例 15

测试/api/questions/{id}接口delete方法。

1) 前置条件

登陆。

2) 测试输入

对/api/questions/{id}发送delete请求。

3) 预期结果

数据库删除指定id的问题。

## 测试用例 16

测试/api/questions/{id}/follow接口get方法。

1) 前置条件

登陆。

2) 测试输入

对/api/questions/{id}/follow发送get请求。

3) 预期结果

返回指定id问题的当前用户是否关注。

## 测试用例 6

测试/api/questions/{id}/follow接口post方法。

1) 前置条件

登陆。

2) 测试输入

对/api/questions/{id}/follow发送post请求。

3) 预期结果

数据库记录当前用户关注指定id问题。

## 测试用例 7

测试/api/questions/{id}/follow接口delete方法。

1) 前置条件

登陆。

2) 测试输入

对/api/questions/{id}/follow发送delete请求。

3) 预期结果

数据库删除当前用户关注指定id问题。

## 测试用例 11

测试/api/questions/{id}/report接口post方法。

1) 前置条件

登陆。

2) 测试输入

对/api/questions/{id}/report发送post请求，包含itemId、itemType、reason参数。

3) 预期结果

数据库记录该用户对该问题的举报。

## 测试用例 20

测试/api/articles接口get方法。

1) 前置条件

登陆。

2) 测试输入

对/api/articles发送get请求，包含page，size参数。

3) 预期结果

收到以给定size进行分页的第page页文章数据。

## 测试用例 21

测试/api/articles/{id}接口get方法。

1) 前置条件

登陆。

2) 测试输入

对/api/articles/{id}发送get请求。

3) 预期结果

收到id为指定参数的文章数据。

## 测试用例 22

测试/api/articles接口post方法。

1) 前置条件

登陆。

2) 测试输入

对/api/articles发送post请求，包含title、summary、content参数。

3) 预期结果

数据库记录以title为标题、以summary为简介、以content为内容的新文章。

## 测试用例 23

测试/api/articles/{id}接口delete方法。

1) 前置条件

登陆。

2) 测试输入

对/api/articles/{id}发送delete请求。

3) 预期结果

数据库删除指定id的文章。

## 测试用例 24

测试/api/articles/{id}/like接口get方法。

1) 前置条件

登陆。

2) 测试输入

对/api/articles/{id}/like发送get请求。

3) 预期结果

返回指定id文章的当前用户是否点赞。

## 测试用例 25

测试/api/articles/{id}/like接口post方法。

1) 前置条件

登陆。

2) 测试输入

对/api/articles/{id}/like发送post请求。

3) 预期结果

数据库记录当前用户点赞指定id文章。

## 测试用例 26

测试/api/articles/{id}/like接口delete方法。

1) 前置条件

登陆。

2) 测试输入

对/api/articles/{id}/like发送delete请求。

3) 预期结果

数据库删除当前用户点赞指定id文章。

## 测试用例 27

测试/api/articles/{id}/favorite接口get方法。

1) 前置条件

登陆。

2) 测试输入

对/api/articles/{id}/favorite发送get请求。

3) 预期结果

返回指定id文章的当前用户是否收藏。

## 测试用例 28

测试/api/articles/{id}/favorite接口post方法。

1) 前置条件

登陆。

2) 测试输入

对/api/articles/{id}/favorite发送post请求。

3) 预期结果

数据库记录当前用户收藏指定id文章。

## 测试用例 29

测试/api/articles/{id}/favorite接口delete方法。

1) 前置条件

登陆。

2) 测试输入

对/api/articles/{id}/favorite发送delete请求。

3) 预期结果

数据库删除当前用户收藏指定id文章。

## 测试用例 30

测试/api/articles/{id}/report接口post方法。

1) 前置条件

登陆。

2) 测试输入

对/api/articles/{id}发送post请求，包含itemId、itemType、reason参数。

3) 预期结果

数据库记录该用户对该文章的举报。

## 测试用例 31

测试/api/moments接口get方法。

1) 前置条件

登陆。

2) 测试输入

对/api/moments发送get请求，包含page，size参数。

3) 预期结果

收到以给定size进行分页的第page页点滴数据。

## 测试用例 32

测试/api/moments/{id}接口get方法。

1) 前置条件

登陆。

2) 测试输入

对/api/moments/{id}发送get请求。

3) 预期结果

收到id为指定参数的点滴数据。

## 测试用例 33

测试/api/moments接口post方法。

1) 前置条件

登陆。

2) 测试输入

对/api/moments发送post请求，包含content、medias参数。

3) 预期结果

数据库记录以content为内容·、以medias为媒体信息的新点滴。

## 测试用例 34

测试/api/moments/{id}接口delete方法。

1) 前置条件

登陆。

2) 测试输入

对/api/moments/{id}发送delete请求。

3) 预期结果

数据库删除指定id的点滴。

## 测试用例 35

测试/api/moments/{id}/like接口get方法。

1) 前置条件

登陆。

2) 测试输入

对/api/moments/{id}/like发送get请求。

3) 预期结果

返回指定id点滴的当前用户是否点赞。

## 测试用例 36

测试/api/moments/{id}/like接口post方法。

1) 前置条件

登陆。

2) 测试输入

对/api/moments/{id}/like发送post请求。

3) 预期结果

数据库记录当前用户点赞指定id点滴。

## 测试用例 37

测试/api/moments/{id}/like接口delete方法。

1) 前置条件

登陆。

2) 测试输入

对/api/moments/{id}/like发送delete请求。

3) 预期结果

数据库删除当前用户点赞指定id点滴。

## 测试用例 38

测试/api/moments/{id}/favorite接口get方法。

1) 前置条件

登陆。

2) 测试输入

对/api/moments/{id}/favorite发送get请求。

3) 预期结果

返回指定id点滴的当前用户是否收藏。

## 测试用例 39

测试/api/moments/{id}/favorite接口post方法。

1) 前置条件

登陆。

2) 测试输入

对/api/moments/{id}/favorite发送post请求。

3) 预期结果

数据库记录当前用户收藏指定id点滴。

## 测试用例 40

测试/api/moments/{id}/favorite接口delete方法。

1) 前置条件

登陆。

2) 测试输入

对/api/moments/{id}/favorite发送delete请求。

3) 预期结果

数据库删除当前用户收藏指定id点滴。

## 测试用例 41

测试/api/moments/{id}/report接口post方法。

1) 前置条件

登陆。

2) 测试输入

对/api/moments/{id}发送post请求，包含itemId、itemType、reason参数。

3) 预期结果

数据库记录该用户对该点滴的举报。