SSC

系统接口定义

版 本 历 史

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 提交日期 | 版本号 | 作 者 | 修 改 内 容 | 备注 |
| 2017-7-6 | 1.0 |  | 初始版本 |  |

术语缩写与解释

|  |  |  |  |
| --- | --- | --- | --- |
| 缩 写 | 解 释 | 缩 写 | 解 释 |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |

目 录

1. 总述 11

1.1 网络结构 11

1.2 接口协议 12

1.3 接口统一约定 12

1.4 身份认证安全设计 13

1.4.1 总体约定 13

1.4.2 注册及修改密码 13

1.4.3 认证登录 13

1.4.4 服务登录及通信 14

1.5 文件存储及传输 14

2. UPDATE接口 15

2.1 检测前端版本 15

2.2 文件下载 16

3. AUTH接口 16

3.1 获取用户基本信息 16

3.2 获取短信验证码 17

3.3 添加用户 17

3.4 用户注册 18

3.5 认证登录 19

3.6 获取用户信息 20

3.7 修改用户信息 21

3.8 重置用户密码 22

3.9 修改用户手机号码 22

4. IM接口 23

4.1 服务登录 23

4.2 消息上行 23

4.3 消息下行 24

4.4 获取聊天记录数量 25

4.5 获取聊天记录 26

5. BUSINESS共用接口 27

5.1 服务登录 27

5.2 文件上传 27

5.3 文件下载 28

5.4 获取家庭成员列表或个人基本信息 28

5.4 获取个人健康档案列表 29

5.5 获取交流圈列表 30

5.6 获取交流圈留言列表 31

5.7 发表交流圈留言 32

5.8 删除交流圈留言 33

5.9 获取预约挂号列表 33

5.10 修改预约挂号 35

5.11 获取计划免疫列表 37

5.12 修改计划免疫 38

5.13 获取下级行政区列表 39

5.14 意见反馈 39

5.15 获取内部推送消息（暂停使用） 40

6. BUSINESS健教专干端接口 41

6.1 获取任务计划 41

6.2 添加任务计划 42

6.3 获取活动类型状态 42

6.4 获取通知消息 43

6.5 获取备忘列表 45

6.6 修改备忘 46

6.7 获取网格地图 46

6.8 提交网格建档状态 48

6.9 添加新住址 48

6.10 修改家庭成员或个人基本信息 49

6.11 家庭成员迁出 51

6.12 获取可迁入人员 51

6.13 家庭成员迁入 52

6.14 获取健康档案目录 53

6.15 获取健康档案界面 54

6.16 提交居民健康档案 55

6.A01 提交居民健康影像档案 55

6.A02 获取居民所属人群 56

6.A03 添加居民所属人群 57

6.17 获取活动信息 58

6.18 发起活动 60

6.A04 添加参加活动的人员 61

6.19 活动签到或登记 62

6.20 进入活动总结 63

6.21 提交活动总结 63

6.22 获取所辖居民 64

6.23 修改交流圈 65

6.24 获取家庭医生签约列表 66

6.25 获取家庭医生签约信息 67

6.26 修改家庭医生签约信息 69

7. BUSINESS居民端接口 70

7.A1 获取主页轮播图 70

7.A2 获取主页文章列表 71

7.A3 获取公司简介 71

7.1 获取通知消息 73

7.2 回复通知消息 74

7.3 获取预约挂号医院、科室及常见病种列表 74

7.4 获取预约挂号医师列表（暂停使用） 76

7.5 授权查看个人健康档案 76

7.6 获取对应健教专干 77

7.7 获取家庭医生签约申请 78

7.8 提交家庭医生签约申请 78

7.9 提交测一测结果 79

8. BUSINESS LIS数据同步接口 80

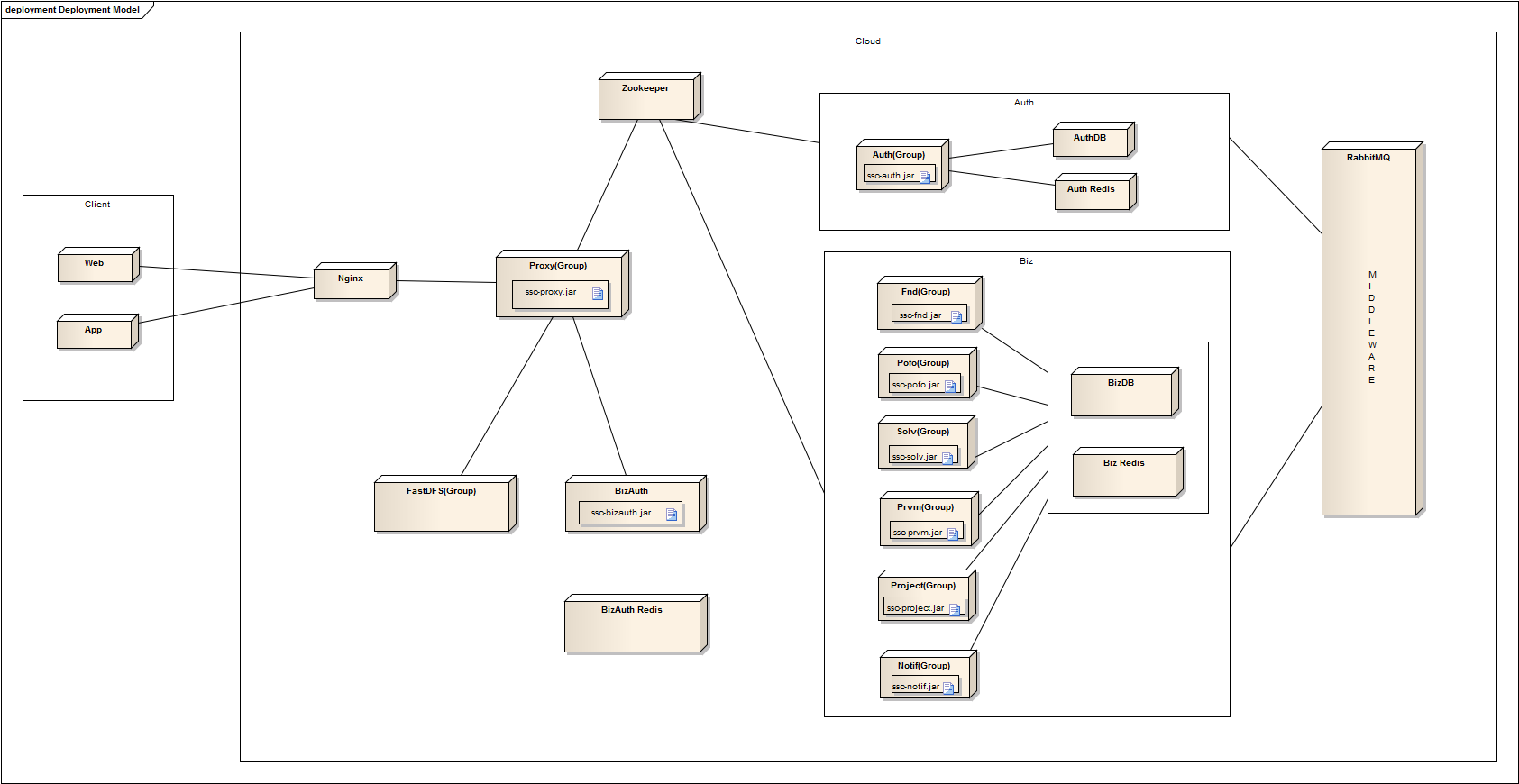
8.1 提交库贝尔iOSS LIS检查结果数据 80

9. 错误码定义 81

# 1. 总述

## 1.1 网络部署图

系统网络结构如下：



包含以下部分, **系统模块:**

1. Proxy(Group)

Proxy模块用于接收HTTP请求, 并调用后台DUBBO服务

**注:** 使用(Group)标志均为集群部署, 下同.

1. Auth(Group)

登录认证服务, 实现用户登录.

1. BizAuth(Group)

业务认证模块, 实现服务登录 和 通讯认证请求

1. FND

业务基础模块, 提供系统角色 / 基础数据 / 配置等服务

1. POFO

Post / Follow的简称, 此模块实现post / 互相关注 / 评论等功能

1. SOLV

SOLV模块实现问答系统功能, 促进用户之间的互助 与 交流

1. PRVM

Private Message(私信)模块, 提供用户之间发送私下信息的服务

1. Notif

Notification的简写, 聚合了系统各项通知信息

1. Project

N/A

**系统还包含一下基础组件:**

1. Nginx

前端反向代码, 作为负载均衡使用.

1. Zookeeper

Dubbo RPC服务注册中心.

1. RabbitMQ

消息中间件, 保障系统的灵活 / 可扩展性

1. MySQL

系统采用MySQL作为持久化数据库

1. Redis

用于实现缓存 / 分布式ID生成器 / 分布式锁等功能

1. FastDFS

开源分布式文件存储服务器，适合比较小文件存储

## 1.2 接口约定

### 1.2.1 协议及数据

系统外部接口协议采用HTTP协议，使用GET / POST方法。

数据传输按照请求类型采用如下格式:

|  |  |
| --- | --- |
| POST方法 | JSON格式请求体 |
| GET方法 | URL参数，示例：/xxx?param1=val1&param2... |

接口返回数据统一采用JSON格式，可包含以下属性：

|  |  |
| --- | --- |
| ret\_code | 状态码标志，1成功 2失败 |
| ret\_msg | 请求失败时返回的消息 |
| ret\_data | 请求成功时返回的数据 |

### 1.2.2 Lookup类型

对于Lookup/Enum类型的字段一律返回code，不返回value。比如：用户状态1. active2. suspend 返回1、2，不返回具体文本

### 1.2.3 文件URL

文件URL一律由服务端拼接之后返回全路径。

## 1.3身份认证设计

身份认证体系借鉴和简化自Kerberos V4协议。

### 1.3.1总体约定

1）本系统所有界面输入的密码，在系统内一律使用：

**HEX(SHA256("SR&09$UINxA09iWY+实际输入的密码+P0(UJkUI9#M0kI7ZEaL"))**

以下简称“变换后的密码”。

如需在前端保存密码，一律对“变换后的密码”再变换加密保存，再变换的算法各端自定。

2）本系统身份认证涉及的所有AES加密，均使用如下方式：

**"AES/CFB/NoPadding"，iv="O09&YA)We3qMI5EX"**

3）一些说明

|  |  |
| --- | --- |
| 源类型 | Web ／ iOS ／ Android |
| access\_token | 客户端发起请求时用作身份验证 |
| auth\_token | 用于实现web端·记住密码·功能 |

### 1.3.2 注册及修改密码

凡需传输密码，前端须对“变换后的密码”做AES加密再发送给Authentication Server，即：

**Base64(AES("变换后的密码", KEY("72F3897670FE064B0C9A7BAE669F0772")))**

Authentication Server收到后，解密存入数据库。

### 1.3.3 认证登录

1）前端向认证服务发送认证登录请求，请求中包含登录名或者用户邮箱，以及如下认证字串：

**Base64(AES("****32Byte随机密钥+源类型/登录名", KEY("变换后的密码")))**

其中：

随机密钥由前端在每次登录时随机生成，认证服务用它来加密登录响应。客户端随后使用该密钥进行揭秘。

2）认证服务收到请求后，从数据库中读取“变换后的密码”，作为密钥解密上述密文。

3）如果认证登录成功，响应中包含认证结果字串：

**Base64(AES("Access Token(uuid)+登录名", KEY("32Byte随机密钥")))**

~~其中：~~

~~传输密钥由认证服务在每次登录时重新分配，可在后续前后端的通信中用来加密。~~

~~登录成功响应中还包含服务票据字串：~~

**~~Base64(AES("32Byte传输密钥+源类型/登录名+登录ID(uuidv4)", KEY("32Byte服务密钥")))~~**

~~其中：~~

~~服务密钥为各服务之间统一的固定密钥，只在服务端保存，对前端保密。~~

4）前端收到认证登录响应后，用随机密钥解密认证结果，核对登录名。如果登录成功，保存user\_login ／ access\_token ／ auth\_token(可选)。

## 1.4 文件存储及传输

服务端文件系统采用**FastDFS**，配置为Proxy访问

# 2. AUTH

## 2.1账号注册

|  |  |  |
| --- | --- | --- |
| URL | /ssc/auth/signup | POST |
| 请求 | {  "login\_name": "xx" // 登录名  "user\_mail": "xx" // 邮箱  "passwd\_str": "xx" // 用户密码  "position\_type": 0 // 职责类型, 1 开发 2 测试 3 产品 4 设计 5 运维 etc  } | 1、登录名 和 邮箱须唯一  2、passwd\_str需要变换之后再加密，参考《身份认证设计》  3、发送注册邮件（异步），账户无需确认可以使用 |
| 响应 | {  "ret\_code":0 //返回码  "ret\_msg":"xx" //可选，返回错误时的错误描述  "ret\_data": //响应数据  {  "user\_login":"xxx" //用户登录名  "auth\_ret\_str":"xx" //认证结果密文串  "svc\_ticket\_str":"xx" //服务票据密文串  }  } | 10-16: 添加position\_type |
| Secured | **No** | |

## 2.2认证登录

|  |  |  |
| --- | --- | --- |
| URL | /ssc/auth/login | POST |
| 请求 | {  "user\_name": "xx" // 用户名  "login\_auth\_str": "xx" // 认证登录密文  "remember\_me": 0 // 可选，是否保持登录 0 否 1 是 仅用于Web端  } | 1、user\_name可以是登录名 或者邮箱 |
| 响应 | {  "ret\_code":0 //返回码  "ret\_msg":"xx" //可选，返回错误时的错误描述  "ret\_data": //响应数据  {  "user\_login":"xxx" //用户登录名  "auth\_ret\_str":"xx" //认证结果密文串  ~~"svc\_ticket\_str":"xx" //服务票据密文串~~  }  } | user\_login - 登录名 |
| Secured | **No** | |

## 2.3忘记密码

|  |  |  |
| --- | --- | --- |
| URL | /ssc/auth/forgetpwd | POST |
| 请求 | {  "user\_mail":"xx" //用户邮箱  } | 收到请求之后发送重置密码邮件  1、生成唯一标志码标志此次密码重置 |
| 响应 | {  "ret\_code":0 //返回码  "ret\_msg":"xx" //可选，返回错误时的错误描述  } |  |
| Secured | **No** |  |

## 2.4 重置密码

|  |  |  |
| --- | --- | --- |
| URL | /ssc/auth/resetpwd | POST |
| 请求 | {  "user\_mail":"xx"//用户邮箱  "new\_pwd":"xx" //新密码  ~~"token\_reset\_pwd": "xxx" //重置密码token~~  "verify\_code": "xxx" //验证码  } | ~~通过收到邮件中的URL重置密码~~  通过邮件收到的验证码重置用户密码 |
| 响应 | {  "ret\_code":0 //返回码  "ret\_msg":"xx" //可选，返回错误时的错误描述  } |  |
| Secured | **No** |  |

# 4. FND

## ~~4.1 获取用户列表~~

|  |  |  |
| --- | --- | --- |
| URL | /ssc/fnd/user/list | GET |
| 请求 | 参数：  q=xx // 查询关键字，可选  user\_status= 0 // 1. active 2. suspend  begin=0 // 起止  count=20 // 条数  access\_token=xxx | 1、q参数可选，如果不提供则查询所有；提供则按照登录名、用户名以及邮箱模糊查询  2、需判断当前用户是否有此权限 |
| 响应 | {  "ret\_code":0 //返回码  "ret\_msg":"xx" //可选，返回错误时的错误描述  "ret\_data": //响应数据  [  {  // 账号信息  "user\_id":0 //用户ID  "user\_login":"xx" //用户登录名  "user\_mail":"xx" //用户邮箱  "mail\_verified":0 // 0 未验证 1 已验证  "user\_status":0 // 1. active 2. suspend  "signup\_date": 0L // 注册日期，时间戳类型  // 用户信息  "user\_name":"xx" //用户名  }  ...  ]  } |  |
| Secured | **Yes** | |

## 4.2 获取用户信息

|  |  |  |
| --- | --- | --- |
| URL | /ssc/fnd/user | GET |
| 请求 | 参数列表：  user\_login="xx"  access\_token="xx"  user\_login\_get="xx" // 可选, 如果不传则获取自己的 | 要点：  ~~1、user\_login可以是当前用户自己或者其他人的~~ |
| 响应 | {  "ret\_code":0 //返回码  "ret\_msg":"xx" //可选，返回错误时的错误描述  "ret\_data": //响应数据  {  //账号信息  "user\_login":"xx" //用户登录名  "user\_mail":"xx" //用户邮箱  "mail\_verified":0 // 0 未验证 1 已验证  "user\_status":0 // 1. active 2. suspend  "signup\_date": 0L // 注册日期，时间戳类型  // 用户信息  "user\_name":"xx" //用户名  "user\_profile":"xxxx" // 用户头像 fdfs地址  "profile\_status": 0 // 头像状态 1. 待审核2. 审核通过 3. 拒绝  }  } |  |
| Secured | **Yes** | |

## 4.3 修改用户信息

|  |  |  |
| --- | --- | --- |
| URL | /ssc/fnd/user/update | POST |
| 请求 | {  "user\_login":xxx  "access\_token":xxx  ~~"user\_id\_upd":0 //修改的用户ID，可选，修改自己的不携带~~  // 账号信息  ~~"user\_login":"xx" //用户登录名~~  ~~"user\_mail":"用户邮箱" //用户邮箱~~  ~~"user\_status":0 //用户状态 1. active 2. suspend~~  ~~"old\_passwd": "xxx" //旧密码~~  ~~"new\_passwd": "xxx" //新密码~~  // 用户信息  "user\_name":"xxx" //用户名  "user\_profile":"xx" //用户头像 // 1. 待审核2. 审核通过3. 拒绝  ~~"profile\_status":"xx" //头像审批状态~~  } | 修改自己的用户信息:  1、可修改~~登录名、~~用户名、用户头像~~、邮箱地址和密码~~  ~~2、修改邮箱 和 密码，需要输入当前密码~~  ~~管理员：~~  ~~1、可修改头像状态、停用启用用户~~ |
| 响应 | {  "ret\_code":0 //返回码  "ret\_msg":"xx" //可选，返回错误时的错误描述  } |  |
| Secured | **Yes** |  |

## 4.4标签分组

|  |  |  |
| --- | --- | --- |
| URL | /ssc/fnd/tags | POST |
| 请求 | 参数列表：  position\_type=0 // 职位类型 |  |
| 响应 | {  "ret\_code":0 //返回码  "ret\_msg":"xx" //可选，返回错误时的错误描述  "ret\_data":[  {  "tag\_group": "xx"  "tag\_list": [  {  "tag\_id": 0  "tag\_text": "xx"  "tag\_hits": 0  }  ]  }  ]  } | 0927: add  1016:  添加position\_type |
| Secured | **No** |  |

## 4.5 设置用户标签

|  |  |  |
| --- | --- | --- |
| URL | /ssc/fnd/user/settag | POST |
| 请求 | {  "user\_login":xxx  "access\_token":xxx  "tag\_list":[xxx] // 标签ID列表  } |  |
| 响应 | {  "ret\_code":0 //返回码  "ret\_msg":"xx" //可选，返回错误时的错误描述  } | 0927: add |
| Secured | **Yes** |  |

## 4.6 文件上传

|  |  |  |
| --- | --- | --- |
| URL | /ssc/fnd/file/upload | POST |
| 请求 | 文件file内容  {  "user\_login":"xx"  "access\_token":"xx"  } | 1、上传图片时生成图片的缩略图（异步） |
| 响应 | {  "ret\_code":0 //返回码  "ret\_msg":"xx" //可选，返回错误时的错误描述  "ret\_data": //响应数据  {  "file\_url":"xx" //文件URL  }  } |  |
| Secured | **Yes** |  |

# 5. POFO

## 5.1发帖

|  |  |  |
| --- | --- | --- |
| URL | /ssc/pofo/post/create | POST |
| 请求 | {  "user\_login":"xx"  "access\_token":"xx"  "post\_content": "xx" // 帖子内容，可选  "post\_id\_refer": 0L // 转帖引用的帖子  "comment\_id\_refer": 0L // 引用评论id  "post\_imgs": [  "xx"  ...  ]  "notify\_users": [  "xx" // user\_login列表  ...  ]  } | 此接口为发帖、转帖、引用帖子公用接口，可以引用评论发帖；评论本身不作为帖子  1、两种类型帖子  *i*、{xx}\_id\_refer参数为空，普通帖子  *ii*、{xx}\_id\_refer非空为转帖, content/ imgs可选  提交后异步生成帖子  hashtag  2、只允许最多6张图片 |
| 响应 | {  "ret\_code":0 //返回码  "ret\_msg":"xx" //可选，返回错误时的错误描述  "ret\_data": {  "post\_id": 0 // 返回帖子id  }  } |  |
| Secured | **Yes** |  |

## 5.2 关注 / 取消关注

|  |  |  |
| --- | --- | --- |
| URL | /ssc/pofo/follow | POST |
| 请求 | {  "user\_login":"xx"  "access\_token":"xx"  "user\_login\_follow": "xx" // 被关注用户  "action\_type":0 // 1 关注 2 取消关注  } |  |
| 响应 | {  "ret\_code":0 //返回码  "ret\_msg":"xx" //可选，返回错误时的错误描述  } |  |
| Secured | **Yes** |  |

## 5.3 列表

|  |  |  |
| --- | --- | --- |
| URL | /ssc/pofo/post/list | GET |
| 请求 | 参数：  user\_login=xx  access\_token=xx  q=xx // 查询关键字，可选  user\_login\_get="xx" // 用户登录名，可选  ~~show\_img=0 // 是否显示图片 0 不显示 1 显示~~  begin=0 // 起止  count=20 // 条数 | 1、首页列表，不带q 和 user\_login参数  2、帖子搜索使用q参数  3、查看指定用户推文使用user\_login参数  以上均按照时间倒序 |
| 响应 | {  "ret\_code":0 //返回码  "ret\_msg":"xx" //可选，返回错误时的错误描述  "ret\_data": [  {  "post\_id": 0L // 帖子id  "user\_name": "xx" // 用户名  "user\_login": "xx" // 登录名  "user\_profile": "xx" // 头像  "post\_type": 0 // 1 帖子 2 转帖 3 引用帖子  "post\_content": "xx" // 帖子内容  "post\_imgs": [ // 帖子图片，可选  "xx"...  ]  "post\_refers": 0 // 转帖数  "post\_likes": 0 // 点赞  "post\_comments": 0 // 帖子评论数  "create\_date": 0L // 帖子发表日期  // 转帖类型为帖子  "post\_id\_refer": 0 // 引用的帖子id  "post\_name\_refer": "xx" // 引用帖子用户名  "post\_login\_refer": "xx" // 引用帖子登录名  "post\_content\_refer": "xx" // 引用帖子的内容  ~~// 转帖类型为评论~~  ~~"comment\_id\_refer": 0// 引用的评论id~~  ~~"comment\_name\_refer": "xx" // 引用评论的用户名~~  ~~"comment\_login\_refer": "xx" // 引用评论的登录名~~  ~~"comment\_content\_refer": "xx"// 引用评论的内容~~  ~~"comment\_name\_reply": "xx"// 评论回复的用户名~~  ~~"comment\_login\_reply": "xx"// 评论回复的用户登录~~  }  ]  } | 修改历史  0920：去掉评论转帖  0920：去掉show\_img  参数 |
| Secured | **Yes** |  |

## 5.4 详情

|  |  |  |
| --- | --- | --- |
| URL | /ssc/pofo/post | GET |
| 请求 | 参数：  user\_login=xx  access\_token=xx  post\_id=0L |  |
| 响应 | {  "ret\_code":0 //返回码  "ret\_msg":"xx" //可选，返回错误时的错误描述  "ret\_data": {  "user\_name": "xx" // 用户名  "user\_login": "xx" // 登录名  "user\_profile": "xx" // 头像  "post\_type": 0 // 1 帖子 2 转帖 3 引用帖子  "post\_content": "xx" // 帖子内容  "post\_refers": 0 // 转帖数  "post\_likes": 0 // 点赞  "post\_comments": 0 // 帖子评论数  "create\_date": 0L // 帖子发表日期  "post\_imgs": [ // 帖子图片  "xx"...  ],  // 转帖类型为帖子  "post\_id\_refer": 0 // 引用的帖子id  "user\_name\_refer": "xx" // 引用帖子用户名  "user\_login\_refer": "xx" // 引用帖子登录名  "post\_content­\_refer": "xx" // 引用帖子的内容  // 转帖类型为评论  "comment\_id\_refer": 0 // 引用的评论id  "comment\_user\_reply": "xx" // 评论回复的用户  }  } | 修改历史  0920：去掉评论转帖 |
| Secured | **Yes** |  |

## 5.5点赞 / 取消点赞

|  |  |  |
| --- | --- | --- |
| URL | /ssc/pofo/post/like | POST |
| 请求 | {  "user\_login":"xx"  "access\_token":"xx"  "post\_id": 0 // 帖子id  "action\_type":0 // 1 点赞 2 取消  } | ~~此接口为点赞、取消点赞共用，如果已经点赞则取消~~ |
| 响应 | {  "ret\_code":0 //返回码  "ret\_msg":"xx" //可选，返回错误时的错误描述  } | Changelog:  0921: 添加action\_type字段 |
| Secured | **Yes** |  |

## 5.6 发表评论

|  |  |  |
| --- | --- | --- |
| URL | /ssc/pofo/comment/create | POST |
| 请求 | {  "user\_login":"xx"  "access\_token":"xx"  "post\_id": 0 // 帖子id, 可选  "comment\_id\_reply": 0 //回复的comment id, 可选  "comment\_content": "xx": //评论内容  "notify\_users": [  "xx" // user\_login列表  ...  ]  } | 评论最多允许两层，第一层为帖子评论，第二层为评论的回复。  1、post\_id不为空为回复帖子；comment\_id不为空回复评论  2、异步发送提醒 |
| 响应 | {  "ret\_code":0 //返回码  "ret\_msg":"xx" //可选，返回错误时的错误描述  } | TODO:  1. notify\_users需要由客户端传吗?  2.增加  comment\_thread |
| Secured | **Yes** |  |

## 5.7评论列表

|  |  |  |
| --- | --- | --- |
| URL | /ssc/pofo/comment/list | GET |
| 请求 | {  "user\_login":"xx"  "access\_token":"xx"  "post\_id": 0 // 帖子id  "comment\_thread": "xxx" // 评论thread  "begin": 0  "count": 0  } |  |
| 响应 | {  "ret\_code":0 //返回码  "ret\_msg":"xx" //可选，返回错误时的错误描述  "ret\_data": {  {  "comment\_id": 0 // 评论id  "user\_name": "xx" // 评论用户名  "user\_login": "xx" // 评论登录名  "user\_profile": "xx" // 评论头像  "comment\_content": "xx" // 评论内容  "notify\_users": ["xx1"...] // 提醒用户  "create\_date": 0L // 评论日期  ~~"comment\_replies": [~~  ~~{~~  ~~"comment\_id": 0 // 评论id~~  ~~"user\_name": "xx" // 评论用户名~~  ~~"user\_login": "xx" // 评论登录名~~  ~~"user\_profile": "xx" // 评论头像~~  ~~"comment\_content": "xx" // 评论内容~~  ~~"notify\_users": ["xx1"...] // 提醒用户~~  ~~"create\_date": 0L // 评论日期~~  ~~}~~  ~~...~~  ~~]~~  ~~}~~  ~~...~~  }  } | 0925:  - 添加comment\_thread参数 |
| Secured | **Yes** |  |

## ~~5.8评论追踪~~

|  |  |  |
| --- | --- | --- |
| URL | /ssc/pofo/comment/track | GET |
| 请求 | 参数：  user\_login=xx  access\_token=xx  comment\_id=0L // 评论id |  |
| 响应 | {  "ret\_code":0 //返回码  "ret\_msg":"xx" //可选，返回错误时的错误描述  "ret\_data": {  "post\_data": {  "user\_name": "xx" // 用户名  "user\_login": "xx" // 登录名  "user\_profile": "xx" // 头像  "post\_type": 0 // 1 帖子 2 转帖 3 引用帖子  "post\_content": "xx" // 帖子内容  "post\_refers": 0 // 转帖数  "post\_likes": 0 // 点赞  "post\_comments": 0 // 帖子评论数  "create\_date": 0L // 帖子发表日期  "post\_imgs": [ // 帖子图片  "xx"...  ],  // 转帖类型为帖子  "post\_id\_refer": 0 // 引用的帖子id  "user\_name\_refer": "xx" // 引用帖子用户名  "user\_login\_refer": "xx" // 引用帖子登录名  "post\_content­\_refer": "xx" // 引用帖子的内容  // 转帖类型为评论  "comment\_id\_refer": 0 // 引用的评论id  "comment\_user\_reply": "xx" // 评论回复的用户  },  "comment\_list":[  {  "comment\_id": 0 // 评论id  "user\_name": "xx" // 评论用户名  "user\_login": "xx" // 评论登录名  "user\_profile": "xx" // 评论头像  "comment\_content": "xx" // 评论内容  "notify\_users": ["xx1"...] // 提醒用户  "create\_date": 0L // 评论日期  }  ...  ]  }  } |  |
| Secured | **Yes** |  |

# 6. SOLV

## 6.1 发布问题

|  |  |  |
| --- | --- | --- |
| URL | /ssc/solv/quest/publish | POST |
| 请求 | {  "user\_login":"xx"  "access\_token":"xx"  "quest\_title": "xx" // 问题标题  "quest\_content": "xx" // 问题内容  "quest\_tags": [  "xx1",  ...  ]  "notify\_users": [  "xx" // user\_login列表  ...  ]  } |  |
| 响应 | {  "ret\_code":0 //返回码  "ret\_msg":"xx" //可选，返回错误时的错误描述  "ret\_data": {  "quest\_id": 0 // 返回问题id  }  } | 0926:  1. create->publish |
| Secured | **Yes** |  |

## 6.2答复

|  |  |  |
| --- | --- | --- |
| URL | /ssc/solv/reply | POST |
| 请求 | {  "user\_login":"xx"  "access\_token":"xx"  "quest\_id": 0L //问题id  "reply\_content": "xx" // 答复内容  "notify\_users": [  "xx" // user\_login列表  ...  ]  } |  |
| 响应 | {  "ret\_code":0 //返回码  "ret\_msg":"xx" //可选，返回错误时的错误描述  "ret\_data": {  "reply\_id": 0 // 返回答复id  }  } | 0926:  1. answer/create -> reply  2. |
| Secured | **Yes** |  |

## 6.3评论

|  |  |  |
| --- | --- | --- |
| URL | /ssc/solv/comment | POST |
| 请求 | {  "user\_login":"xx"  "access\_token":"xx"  "quest\_id": 0L // 问题id  "reply\_id": 0L // 答复id  "comment\_reply": 0L //评论id  "comment\_content": "xx" // 评论内容  "notify\_users": [  "xx" // user\_login列表  ...  ]  } | SOLV中的评论分为三种:  1、问题评论  2、答复评论  3、评论的评论  {xx}\_id至少传递一个 |
| 响应 | {  "ret\_code":0 //返回码  "ret\_msg":"xx" //可选，返回错误时的错误描述  "ret\_data": {  "comment\_id": 0 // 返回评论id  }  } |  |
| Secured | **Yes** |  |

## 6.4问题列表

|  |  |  |
| --- | --- | --- |
| URL | /ssc/solv/quest/list | GET |
| 请求 | 参数：  user\_login=xx  access\_token=xx  q="xx" // 查询关键字，可选  tag="xx" // 标签，可选  begin=0 // 起止  count=20 // 条数 | 1、使用tag时不可搜索 |
| 响应 | {  "ret\_code":0 //返回码  "ret\_msg":"xx" //可选，返回错误时的错误描述  "ret\_data": [  {  "quest\_id":0L // 提问id  "user\_login":"xx" // 提问者  "quest\_title":"xx" // 问题标题  "quest\_tags": [  "xx1",  ...  ]  "quest\_plus ":0 // 同样问题的人数  "quest\_answers":0 // 问题答复数  ~~"quest\_replies":0 // 问题评论数~~  "quest\_accepted":0 // 是否有答案  }  ]  } |  |
| Secured | **No** |  |

## 6.5 问题详情

|  |  |  |
| --- | --- | --- |
| URL | /ssc/solv/quest | GET |
| 请求 | 参数：  qid=0 // 问题ID  user\_login=xx  access\_token=xx |  |
| 响应 | {  "ret\_code":0 //返回码  "ret\_msg":"xx" //可选，返回错误时的错误描述  "ret\_data": {  "quest\_id":0L // 提问id  "user\_login":"xx" // 提问者  "quest\_title":"xx" // 问题标题  "quest\_content":"xx" // 问题内容  "quest\_plus":0 // 同样问题的人数  "quest\_accepted":0 // 是否有答案  "quest\_answer\_id":0L // 答案id，可选  "quest\_tags": [  "xx1",  ...  ]  "quest\_comments": [ // 问题评论  {  "comment\_id": 0L //  "user\_login":"xx" // 评论者  "comment\_content":"xx" // 评论内容  "comment\_date":0L // 日期  "notify\_users": ["xx...] // 提醒用户  }  ]  "quest\_answers": [ // 问题答案  {  "ans\_id": 0L //  "user\_login":"xx" // 评论者  "ans\_content":"xx" // 评论内容  "ans\_date":0L // 日期  "ans\_accepted":0L // 此答复是否被接受  "notify\_users": ["xx...] // 提醒用户  "comment\_list": [  {  "user\_login":"xx" // 评论者  "comment\_content":"xx" // 评论内容  "comment\_date":0L // 日期  "notify\_users": ["xx...] // 提醒用户  }  ]  }  ]  }  } |  |
| Secured | **No** |  |

## 6.6接受答复

|  |  |  |
| --- | --- | --- |
| URL | /ssc/solv/accept | POST |
| 请求 | {  "user\_login":"xx"  "access\_token":"xx"  "quest\_id": 0L // 问题id  "ans\_id": 0L // 答复id  } | 1、需判断提问者和当前用户是否为同一个人, 否则不允许此操作 |
| 响应 | {  "ret\_code":0 //返回码  "ret\_msg":"xx" //可选，返回错误时的错误描述  } |  |
| Secured | **Yes** |  |

# 7. PRVM

## 7.1发私信

|  |  |  |
| --- | --- | --- |
| URL | /ssc/prvm/mesg/create | POST |
| 请求 | {  "user\_login":"xx"  "access\_token":"xx"  "to\_user\_login": "xx" //接收消息者  "mesg\_type": 0 //1 文字 2 图片  "mesg\_content": "xx" // 消息内容  } | 1、 |
| 响应 | {  "ret\_code":0 //返回码  "ret\_msg":"xx" //可选，返回错误时的错误描述  "ret\_data": {  "mesg\_id": 0 // 返回消息id  }  } |  |
| Secured | **Yes** | |

## 7.2 私信数量

|  |  |  |
| --- | --- | --- |
| URL | /ssc/prvm/mesg/count | GET |
| 请求 | 参数：  user\_login=xx  access\_token=xx | 1、按照thread最后一条消息发送时间倒叙  2、 |
| 响应 | {  "ret\_code":0 //返回码  "ret\_msg":"xx" //可选，返回错误时的错误描述  "ret\_data": [  {  "to\_user\_login":"xx" //消息接收、发送的对方  "new\_count": 0 // 下行新消息数量  "last\_mesg": "xx" // thread中最后一条消息  "last\_mesg\_type":0 // 最后一条消息类型  "last\_mesg\_date":0L // 最后一条消息发送时间  }  ]  } |  |
| Secured | **Yes** |  |

## 7.3 私信列表

|  |  |  |
| --- | --- | --- |
| URL | /ssc/prvm/mesg/list | GET |
| 请求 | 参数：  user\_login=xx  access\_token=xx  to\_user\_login="xx" // 消息接收、发送的对方  begin=0 // 起止  count=20 // 条数 | 1、分页取最近发送和接收的消息  2、需标志下行消息为已读 |
| 响应 | {  "ret\_code":0 //返回码  "ret\_msg":"xx" //可选，返回错误时的错误描述  "ret\_data": [  {  "mesg\_type": 0 // 1 文字 2 图片  "mesg\_content":"xx" // 消息内容  "is\_up": 0 // 0 消息下行 1 消息上行  "mesg\_status": 0 // 1 未读 2 已读 对于上行消息  "send\_date":0L // 消息发送时间，以服务端时间为准  }  ]  } |  |
| Secured | **Yes** | |

# 8. NOTIF

## 8.1 未读通知数量

|  |  |  |
| --- | --- | --- |
| URL | /ssc/notif/newcount | GET |
| 请求 | 参数：  user\_login=xx  access\_token=xx | 1、 |
| 响应 | {  "ret\_code":0 //返回码  "ret\_msg":"xx" //可选，返回错误时的错误描述  "ret\_data": {  "new\_count": 0 // 未读通知数量  }  } |  |
| Secured | **Yes** |  |

## 8.2通知列表

|  |  |  |
| --- | --- | --- |
| URL | /ssc/notif/list | GET |
| 请求 | 参数：  user\_login=xx  access\_token=xx  begin=0 // 起止  count=20 // 条数 | 1、按照通知发送日期倒序  2、需要标志返回的通知为已读  ***Note***：如果未读通知不在第一页，需要翻页之后才能标记为已读 |
| 响应 | {  "ret\_code":0 //返回码  "ret\_msg":"xx" //可选，返回错误时的错误描述  "ret\_data": {  "notif\_id": 0 // 通知id  "notif\_content": "xx" // 通知内容  "notif\_status": 0 // 1 未读 2 已读  }  } |  |
| Secured | **Yes** |  |