**单点登录成员站点**

**集**

**成**

**说**

**明**

**书**

****

2018年01月

目录

[1、说明（BS端） 3](#_Toc514246539)

[2、 单点登录基本流程（BS端） 3](#_Toc514246540)

[2.1登录流程 3](#_Toc514246541)

[2.2登出流程 4](#_Toc514246542)

[3、伪代码（BS端） 4](#_Toc514246543)

[3.1登录： 4](#_Toc514246544)

[3.2登出： 5](#_Toc514246545)

[4、获取用户凭证接口说明 5](#_Toc514246546)

[4.1请求地址 5](#_Toc514246547)

[4.2参数 5](#_Toc514246548)

[4.3返回数据 6](#_Toc514246549)

[5、验证登录用户 6](#_Toc514246550)

[5.1地址 6](#_Toc514246551)

[5.2参数 6](#_Toc514246552)

[5.3返回参数 6](#_Toc514246553)

[6、登出推送服务说明（BS端） 7](#_Toc514246554)

[6.1地址 7](#_Toc514246555)

[6.2参数 7](#_Toc514246556)

[7、CS端单点登录集成说明 7](#_Toc514246557)

[7.1基本流程 7](#_Toc514246558)

[8、伪代码（CS端） 8](#_Toc514246559)

[8.1登录 8](#_Toc514246560)

[9、cs端从宿主程序获取token服务 8](#_Toc514246561)

[9.1地址 8](#_Toc514246562)

# 说明（BS端）

为了更好地让第三方成员站点更好更快地集成单点登录功能，以此文档作为开发人员集成单点登录的开发指导。

该文档会涉及到开发流程，某些名称的规范等等，比如说，单点登录所用到的令牌名称、用户凭证格式、获取用户凭证方式、验证用户名密码的方式、登录成功返回的数据格式等，在开发过程中请开发人员遵循这一套开发流程逻辑。

# 单点登录基本流程（BS端）

### 2.1登录流程

①用户访问第三方成员站点登录页面

②从url参数中获取令牌（**sso\_token**）

③判断是否存在**sso\_token**

1）：存在，拿到该sso\_token的值，调用集成登录中心web service拿到用户凭证④

2）：不存在，直接跳到单点登录集成登录中心登录页面进行登录操作，登录成功后，跳回到成员网站首页。

④判断用户凭证是否存在

1. ：存在，通过用户凭证调用单点登录中心服务，获取用户信息

2）：不存在，直接跳到单点登录集成登录中心登录页面进行登录操作，登录成功后，跳回到成员网站首页。

⑤直接访问单点登录登录页面，输入用户名密码验证，验证通过，跳转到第三方成员站点列表页面，点击图标跳转到相应站点即可。

### 2.2登出流程

①第三方站点登出操作

1）：本地用户登出

2）：清除网站记录的用户信息（通常是cookie或session）

3）：调用登出服务。

②单点登录中心登出

1）：本地用户登出

2）：清除cookie中**caradigm\_sso\_token**的值

3) ：清除网站记录的用户信息

# 3、伪代码（BS端）

### 3.1登录：

页面初始化

从url中获取sso\_token的值 🡪 token

If token ！= null //浏览器存在token数据

通过集成登录中心服务和token的值获取用户凭证 🡪 cert

通过cert的值获取用户信息🡪userinfo

If userinfo != null //登录成功

根据userinfo返回的数据，设置本地用户信息以及权限信息session[“userinfo”]或cookie[“userinfo”] //具体看成员站点是怎么设计的

跳转：当前站点首页

Else //登录失败

跳转：http://单点登录地址?backurl=当前访问地址

Else //浏览器不存在token数据

跳转：http://单点登录地址?backurl=当前访问地址

### 3.2登出：

登出开始

清除本地用户信息session[“userinfo”]或cookie[“userinfo”]

通知单点登录中心登出：登出服务名称?op=LoginOutPush&token=token

# 4、获取用户凭证接口说明

### 4.1请求地址

|  |  |
| --- | --- |
| 属性 | 值 |
| 协议 | http |
| 请求方式 | POST |
| 地址 | <http://10.2.30.66>:8091/WebService/SSO\_WebService.asmx?op=GetCertByToken |

### 4.2参数

|  |  |  |  |
| --- | --- | --- | --- |
| 参数名 | 类型 | 最大长度 | 是否为空 |
| token | String | 100 | 否 |

### 4.3返回数据

|  |  |
| --- | --- |
| 返回参数格式 | 说明 |
| <?xml version="1.0" encoding="utf-8"?>  <AjaxResult>  <state>200</state>  <message>获取成功</message>  <data>admin</data>  </AjaxResult> | state节点：  200成功；  205请求成功，但未获取到用户信息；  500服务器出现异常。  message节点：调用结果说明  data节点：用户名 |

# 5、验证登录用户

### 5.1地址

|  |  |
| --- | --- |
| 属性 | 值 |
| 协议 | HTTP |
| 请求方式 | POST |
| 地址 | <http://10.2.30.66>:8091/WebService/SSO\_WebService.asmx?op= CheckLoginByCert |

### 5.2参数

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| 参数名 | 类型 | 最大长度 | 是否为空 | 默认值 | 备注 |
| op | String | 50 | 否 | logonSSO | 值固定不变 |
| username | String | 50 | 否 | 空 | 用户名 |
| funccode | String | 50 | 否 | 空 | 登录操作代码,从后台管理系统数据库取得 |
| systemcode | String | 50 | 否 | 空 | 系统代码,后台管理系统获取 |

### 5.3返回参数

|  |  |
| --- | --- |
| 返回参数格式 | 说明 |
| <?xml version="1.0" encoding="utf-8"?>  <AjaxResult>  <state>200</state>  <message></message>  <data> ……</data>  </AjaxResult> | state节点：  200成功；  205请求成功，但未获取到用户信息；  500服务器出现异常。  message节点：调用结果说明  data节点：用户名信息及权限信息 |

data节点字符串信息：

|  |  |
| --- | --- |
| 格式 | 说明 |
| {  "result": "success",  "data": "{}",  "userPermission": "{}"  } | result：调用结果成功与否（success、fail）  data：用户信息，详细见5.3.1  userPermission：用户权限数据，详见5.3.2 |

#### 5.3.1 DATA格式说明

|  |  |
| --- | --- |
| {  "UserID": 1,  "UserName": "admin",  "DisplayName": "系统管理员",  "UserPassword": "",  "UserType": "0",  "UserTypeName": "系统用户", "EmpNo": null,  "PID": null,  "Status": null,  "CreateTime": null,  "CreateBy": null,  "UpdateTime": null,  "UpdateBy": null  } | UserID：用户主键  UserName：用户名（工号）  DisplayName：中文名  UserPassword：密码  UserType：用户类型编码  UserTypeName：用户类型名称  EmpNo：工号  PID：患者ID  Status：启用状态  CreateTime：创建时间  CreateBy：创建人  UpdateTime：最后更新时间  UpdateBy：最后更新人 |

#### 5.3.2UserPermission详细说明

|  |  |
| --- | --- |
| 格式 | 说明 |
| {  "UserInfo": null,  "UserRoleInfos": [{}],  "UserRoleFunctionInfos": [{}],  "UserConditionInfos": [ ]  } | UserInfo：用户信息  UserRoleInfo：用户角色列表  UserRoleFunctionInfos：用户角色菜单功能点权限列表  UserConditionInfos：用户的数据权限 |

# 6、登出推送服务说明（BS端）

### 6.1地址

|  |  |
| --- | --- |
| 属性 | 值 |
| 协议 | HTTP |
| 请求方式 | POST,GET |
| 地址 | <http://10.2.30.66>:8091//Handle/SSO\_Handler.ashx |

### 6.2参数

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| 参数名 | 类型 | 最大长度 | 是否为空 | 默认值 | 备注 |
| opt | String | 50 | 否 | LoginOutPush | 值固定不变 |
| token | String | 100 | 否 | 空 | 浏览器存储的令牌 |

# 7、CS端单点登录集成说明

### 7.1基本流程

①用户打开桌面应用程序主窗口

②通过宿主程序提供的服务获取到token

③通过token从集成登录中心的服务（web service）获取用户凭证cert

④通过cert从集成登录中心的服务（web service）获取到用户信息

⑤记录本地用户信息，主窗体加载完成

附：以上②、③、④未获取到成功，给出友好提示：“登录失败，需从集成登录中心重新登录“，并自动退出主窗口。

# 8、伪代码（CS端）

### 8.1登录

主窗体初始化

掉主进程服务获取caradigm\_sso\_token的值 🡪 token

If token ！= null //存在token数据

通过集成登录中心服务和token的值获取用户凭证 🡪 cert

通过cert的值获取用户信息🡪userinfo

If userinfo != null //登录成功

根据userinfo返回的数据，设置客户端用户信息以及权限信息 //具体看成员端是怎么设计的

登录成功：加载主窗口相关组件

Else //获取用户信息失败

给出提示信息：用户登录失败，请从集成登录中心重新登录

Else //获取到token数据失败

给出提示信息：用户登录失败，请从集成登录中心重新登录

### 8.2登出

清空本地用户信息

调用主进程服务清除token数据

# 9、cs端从宿主程序获取token服务

### 9.1地址

|  |  |
| --- | --- |
| 属性 | 值 |
| 协议 | HTTP |
| 请求方式 | GET |
| 地址 | http://localhost:9000/service1/GetToken |

### 9.2 返回数据说明

|  |  |
| --- | --- |
| 格式 | 说明 |
| \"\" | 注意返回的字符串，有时可能本身就是双引号。 |

# 10、cs端从宿主程序清除token服务

### 10.1地址

|  |  |
| --- | --- |
| 属性 | 值 |
| 协议 | HTTP |
| 请求方式 | GET |
| 地址 | http://localhost:9000/service1/DeleteToken |
| 返回值 | 无 |

附：其他所使用的服务请参考4、5服务说明。