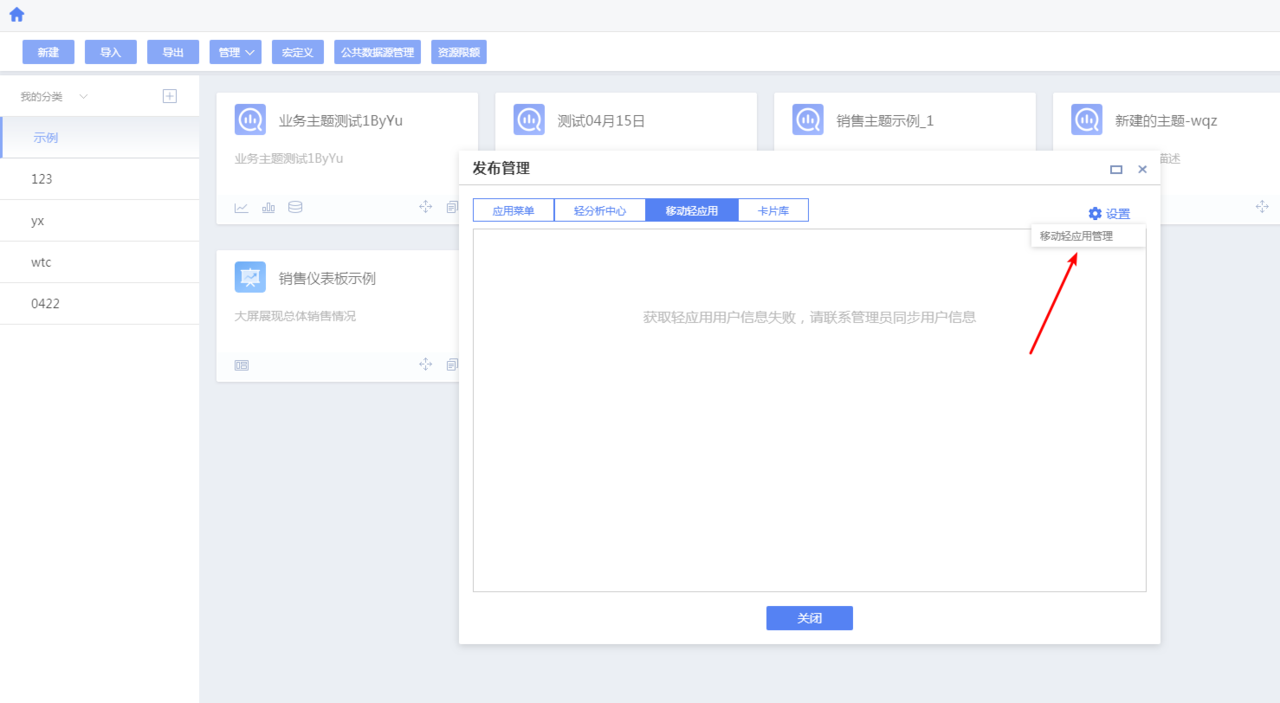
**星空平台轻分析集成移动办公平台配置文档**

轻分析移动轻应用，用于为数据分析、数据斗方和仪表板提供移动端展现能力，支持在多种移动办公平台使用。对于标准产品已经集成的移动办公平台（云之家、企业微信、钉钉），请参考配置文档<https://q.kingdee.com/support/index.html?version=1.0&docId=Tkcaa683n>；对于其他移动办公平台，需以二次开发的方式获取轻分析移动端的H5页面链接，详细教程请阅读本文。

# 配置单点登录规则

## 添加移动办公平台

轻分析管理员进入轻分析主题管理首页，点击管理---->发布管理---->移动轻应用---->设置---->移动轻应用管理



点击添加移动办公平台

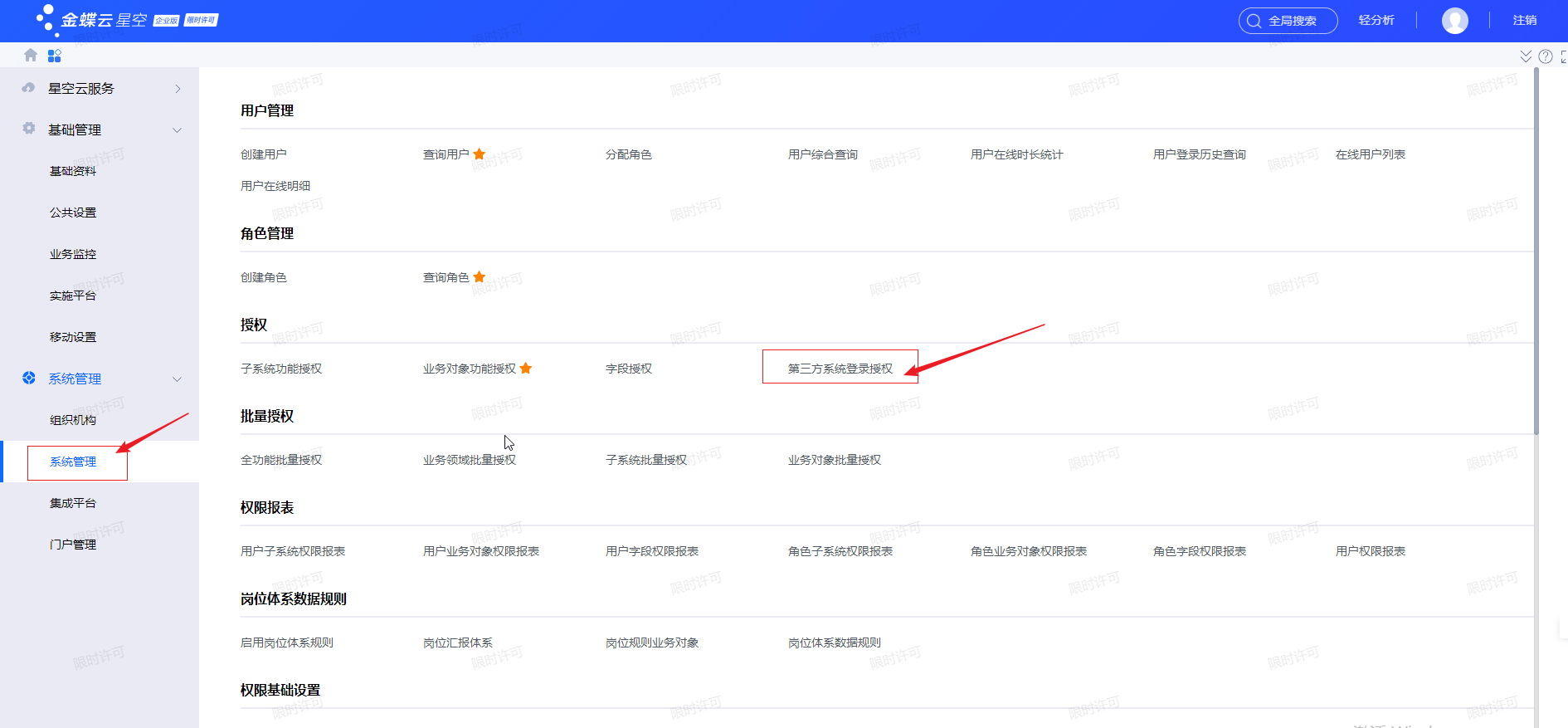


输入移动办公平台ID、用户标识，上传移动办公平台图标后点击确定保存。移动办公平台ID由产品研发团队颁发，获取方式见附录；用户标识与下文[第二步](#_获取accessToken接口)获取accessToken的请求参数appUserType的值所对应。

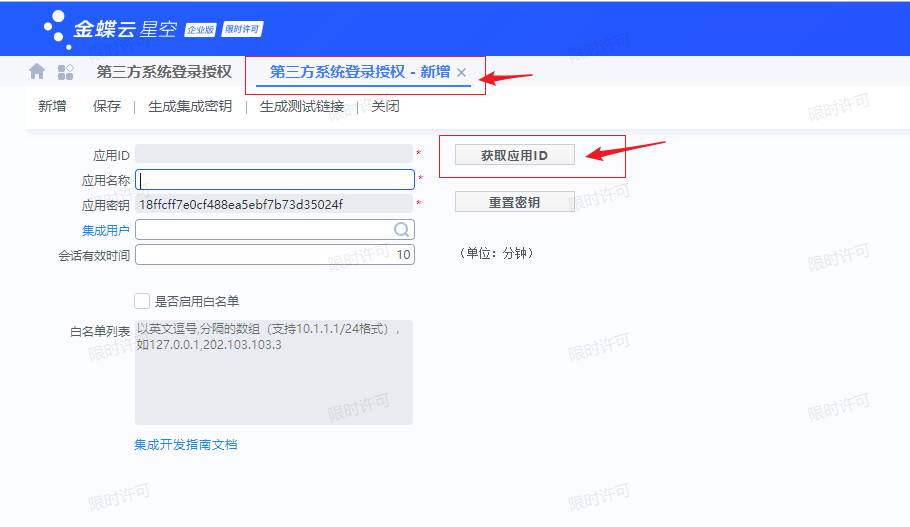


## 星空新增第三方系统单点登录信息

管理员登录星空，点击系统管理，第三方系统登录授权。

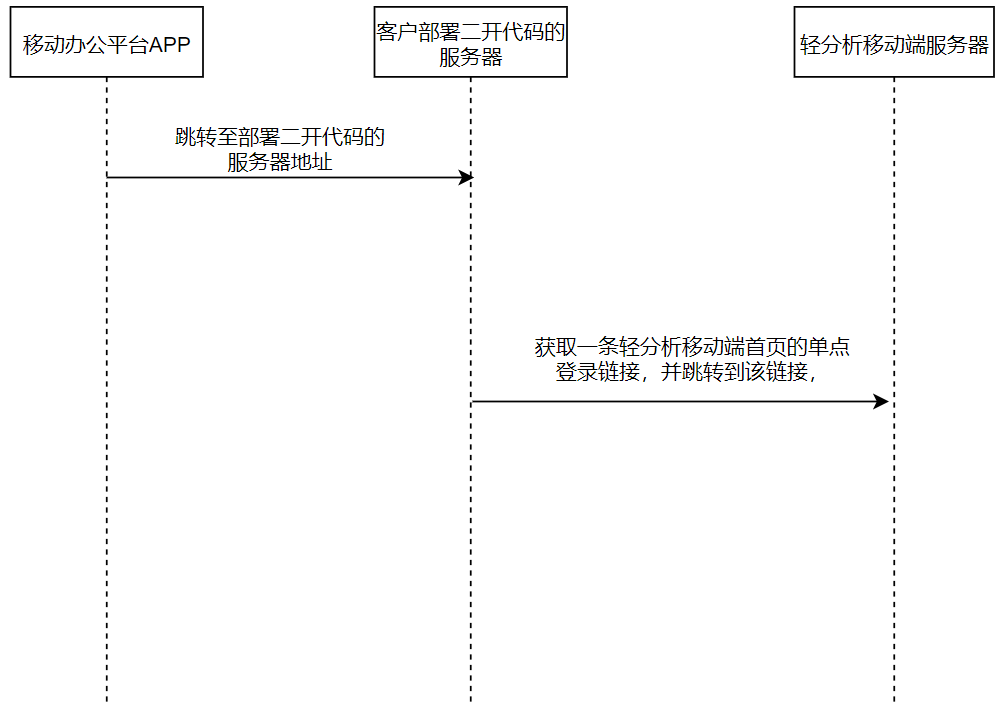


点击新增，获取应用ID，按照提示获取星空应用ID，星空应用密钥，填写应用名称。



# 通过二次开发的方式获取轻分析移动端的H5页面链接

为了在移动办公平台APP上能看到发布到移动轻应用的卡片，需要通过二次开发来获取轻分析移动端的H5页面链接，二次开发的代码部署在客户自己的服务器上。如图所示，在移动办公平台APP上跳转至客户部署二开代码的服务器地址，再重定向到轻分析移动端H5页面链接。



接下来讲述通过二次开发的方法如何调用API接口构建一条单点登录到轻分析移动端H5页面的链接。

## 获取accessToken接口

1. 服务器地址

https://q.kingdee.com/qing-lightapp/webApi/getAccessToken

1. 请求方式：POST
2. 参数说明

|  |  |  |  |
| --- | --- | --- | --- |
| 参数说明 | 必选 | 类型 | 说明 |
| appId | 是 | string | 移动办公平台id，由轻分析产品研发团队颁发 |
| timestamp | 是 | string | Unix时间戳，单位：秒，形如：1552879016 |
| sign | 是 | string | 签名串，除sign外，其他参数均参与签名，签名规则：参数名按字典顺序，把参数值（不含参数名）连接起来，然后md5生成的32位小写字符串，如：  md5(appId+appSecret+cipher+timestamp)，不含加号 |
| cipher | 是 | string | 请求业务数据集（json格式数据），是把json数据经过base64加密的字符串，另外，base64密文字符串还需要再进行utf8格式的urlencode处理。注意：参与签名时，需去掉urlencode |

**注：appId和appSecret获取方式见附录**

1. cipher参数说明(json数据):

|  |  |  |  |
| --- | --- | --- | --- |
| 参数说明 | 必选 | 类型 | 说明 |
| appUserId | 是 | string | 产品端的用户标识（用户名） |
| appUserType | 是 | string | 产品端的用户标识类型（UserName），与appUserId的值相关。 |

cipher参数格式：

{

" appUserId ":"xxxxxxxxxxxx",

"appUserType":"UserName"

}

**注：此cipher参数值(json)，需base64加密且密文要经过urlencode处理**

1. 请求示例

POST /qing-lightapp/webApi/getAccessToken HTTP/1.1

Host: [serverUrl]

Content-Length: xxx

Content-Type: application/x-www-form-urlencoded

appId=xxx&timestamp=xxx&sign=xxx&cipher=xxx

1. 返回参数说明（json格式数据）

返回值（失败）

{

"errorCode": "XX",

"errorMessage": "XXX"

}

返回值（成功）

{

“data”:

{

“accessToken”: “xxx”

}

}

**注：accessToken和用户绑定，两个小时有效，每次调用如果当前用户已有accessToken并没过期则自动延期，有效期重新变为2小时。没有则创建新的accessToken。以后每次业务系统访问轻应用，都携带accessToken访问。**

## 构造轻分析移动端H5页面单点登录链接

1. H5页面链接

https://q.kingdee.com/qing-lightapp/index.html?accessToken=xxx&appType=thirdparty

1. 参数说明

|  |  |  |  |
| --- | --- | --- | --- |
| 参数说明 | 必选 | 类型 | 说明 |
| accessToken | 是 | string | 登录凭证 |
| appType | 是 | string | APP类型，固定为thirdparty |

**注：二次开发代码可参考附录**

# 附录

## appId（移动办公平台ID）和appSecret获取方式

云之家联系郭建波获取，同时提供星空应用ID、星空应用密钥、产品唯一编码：

（1）[第一步中星空新增第三方系统单点登录信息](#_苍穹新增第三方应用单点登录信息)所获取的星空应用ID和星空应用密钥，用于轻分析移动端与星空平台之间的交互；

（2）产品唯一编码，用于检验appId是否填写正确。在[第一步中添加移动办公平台](#_添加移动办公平台)所在界面，将鼠标移动到第一个红点，当鼠标变为手的形状，按下鼠标左键的同时按下[ctrl]键，下方便会出现产品唯一编码。



## 二次开发代码例子

@RequestMapping("/login")  
public void login(HttpServletRequest request, HttpServletResponse response) throws IOException {  
 String serverUrl = "https://q.kingdee.com/qing-lightapp/";  
 String getAccessTokenUrl = serverUrl + "webApi/getAccessToken";  
  
 String appId = "100002"; // 轻分析产品研发团队颁发的移动办公平台ID，根据实际而定  
 String appSecret = "8a94f5e6-6aa3-47ff-90fc-0f054b7e0346"; // 轻分析产品研发团队颁发的移动办公平台密钥，根据实际而定  
 String timestamp = String.*valueOf*(System.*currentTimeMillis*() / 1000);  
 String appUserId = "张三"; // 用户名  
 String appUserType = "UserName"; // 用户标识类型(UserName)   
  
 JSONObject jsonObject = new JSONObject();  
 jsonObject.put("appUserId", appUserId);  
 jsonObject.put("appUserType", appUserType);  
 String jsonStr = jsonObject.toJSONString();  
 String cipher = Base64.*getEncoder*().encodeToString(jsonStr.getBytes());  
  
 StringBuilder sb = new StringBuilder();  
 sb.append(appId).append(appSecret).append(cipher).append(timestamp);  
 String sign = DigestUtils.*md5DigestAsHex*(sb.toString().getBytes());  
 cipher = URLEncoder.*encode*(cipher, "UTF-8");  
  
 Map<String, Object> formMap = new HashMap<>();  
 formMap.put("cipher", cipher);  
 formMap.put("appId", appId);  
 formMap.put("timestamp", timestamp);  
 formMap.put("sign", sign);  
  
 String result = HttpRequest.*post*(getAccessTokenUrl)  
 .header("Content-type", "application/x-www-form-urlencoded")  
 .form(formMap)  
 .execute().body();  
  
 System.*out*.println(result);  
 JSONObject accessTokenJson = JSON.*parseObject*(result);  
 String accessToken = accessTokenJson.getJSONObject("data").getString("accessToken");  
  
 String redirectUrl = serverUrl + "index.html?appType=thirdparty&accessToken=" + accessToken;  
 response.sendRedirect(redirectUrl);  
}