# 融助手 API 接口文档

# 0：token参数

从用户登录获取到用户及token信息，示例如下：

|  |
| --- |
| {  "code": 0,  "data": {  "userInfo": {  "accountName": "liuchao",  "cityCode": "SUZHOU",  "cityName": "",  "dprCode": "",  "dprName": "",  "empCode": "AA1611",  "empName": "刘超[AA1611]",  "roleCode": "",  "roleName": ""  },  "token":"Bearer eyJhbGciOiJIUzUxMiJ9.eyJzdWIiOiJBQTE2MTEiLCJkYXRhIjp7ImFjY291bnROYW1lIjoibGl1Y2hhbyIsImVtcENvZGUiOiJBQTE2MTEiLCJlbXBOYW1lIjoi5YiY6LaFW0FBMTYxMV0iLCJjaXR5Q29kZSI6IlNVWkhPVSIsImNpdHlOYW1lIjpudWxsLCJkcHJDb2RlIjpudWxsLCJkcHJOYW1lIjpudWxsLCJyb2xlQ29kZSI6bnVsbCwicm9sZU5hbWUiOm51bGx9LCJleHAiOjE1MzYxMTMwOTksImlhdCI6MTUzNTUwODI5OX0.OoLpECeSs8oAnWiZ9xc7E81UP\_hTSGm7Wy1J\_yjM43lXbu8dvT7MyhaW-1\_FH50oX1kIM-7SPvmsF0hXzvy2gw"  },  "msg": ""  } |

后续请求需将此token带入请求头，这里我们采用比较通用的Http Authorization + Bearer Token的形式，示例如下：

|  |
| --- |
| GET /resource HTTP/1.1  Host: server.example.com  Authorization: Bearer eyJhbGciOiJIUzUxMiJ9.eyJzdWIiOiJBQTE2MTEiLCJkYXRhIjp7ImFjY291bnROYW1lIjoibGl1Y2hhbyIsImVtcENvZGUiOiJBQTE2MTEiLCJlbXBOYW1lIjoi5YiY6LaFW0FBMTYxMV0iLCJjaXR5Q29kZSI6IlNVWkhPVSIsImNpdHlOYW1lIjpudWxsLCJkcHJDb2RlIjpudWxsLCJkcHJOYW1lIjpudWxsLCJyb2xlQ29kZSI6bnVsbCwicm9sZU5hbWUiOm51bGx9LCJleHAiOjE1MzYxMTMwOTksImlhdCI6MTUzNTUwODI5OX0.OoLpECeSs8oAnWiZ9xc7E81UP\_hTSGm7Wy1J\_yjM43lXbu8dvT7MyhaW-1\_FH50oX1kIM-7SPvmsF0hXzvy2gw |

PS：

Bearer Token 的格式：

Bearer XXXXXXXX

可以参考这篇文章：[OAuth 2.0: Bearer Token Usage](https://www.cnblogs.com/XiongMaoMengNan/p/6785155.html)

# 1: 用户登录

## 1.1：请求方法和地址

接口方法：Post 请求地址：api/account/login

## 1.2：请求参数

请求参数： 【JSON格式】

|  |  |  |
| --- | --- | --- |
| Key | Value | 备注 |
| username | 销售工号 | String |
| password | 密码 | String |

请求案例

|  |
| --- |
| http://10.40.3.110:8077/api/account/login  {  "username":"XXX",  "password":"e10adc3949ba59abbe56e057f20f883e"  } |

## 1.3：响应结果

返回：【JSON格式】

|  |
| --- |
| 1:登录成功返回  {  msg: "登录成功"  code: 0  data: {  userInfo:{  accountName: 账号名称  empCode: 用户编号  empName: 用户姓名  cityCode: 城市编号  cityName: 城市名称  dprCode: 部门编号  dprName: 部门名称  roleCode: 职位编号  roleName: 职位名称  **empWorkMobile:销售手机**  **empWechatId：销售微信ID**  }  token：xxxxxxxxxxxxx  }  }  2：登录失败返回  {  msg:"账户或密码错误"  code：1  } |

校验Token失败

token校验错误返回

{

msg:"token校验错误",

code：500,

data: null

}

# 2: 查询客户列表

## 2.1：请求方法和地址

接口方法：Post 请求地址：api/customer/list

## 2.2：请求参数

请求参数：【JSON格式】

|  |  |  |
| --- | --- | --- |
| Key | Value | 备注 |
| customerExeStatus | 客户意向进程 | Int  1 有需求 2 有意向 3 已来访 4已签约 (5外包成交 6 会员成交) 都是成交 |
| pageSize | 每页记录行数 | Int |
| pageIndex | 初始化加载第几页 | Int |
| empCode | 销售工号 | String |
| **refreshWX** | **是否更新微信好友** | **int 1更新 0不更新** |

**注意：**

**1：在数据库中查询客户意向进程**

**全部页面 查询返回的时候 成交信息包含了5和6。此时查询提交参数是 -1**

**但是在单独查询成交页面 查询接口的时候只需要传递查询参数是 5 ，**

**后台会根据5和6去查询**

**2：目前是否更新微信好友这个功能取消了。传递0和传递1是一样的效果。**

**请求案列：**

|  |
| --- |
| **http://10.40.4.112:8077/api/customer/list**  **{**  **"customerExeStatus": 0,**  **"pageSize": 10,**  **"pageIndex": 1,**  **"empCode": "xxxxxxx"**  **}** |

## 2.3：响应结果

返回：Customer对象组成的列表 【JSON格式】

Customer对象如下：

|  |  |  |
| --- | --- | --- |
| Key | Value | 备注 |
| name | 客户姓名 | String |
| hasCars | 是否有车产 | Int 1：代表了有 2 无 |
| hasPolicy | 是否有担保 | Int 1 代表了有 2 代表了无 |
| salary | 月打卡工资 | Double |
| hasHouse | 是否有房产 | int 1 代表了有 2代表了无房产 |
| IsSecuityOrFund | 是否社保公积金 | int  1代表了无社保无公积金  2 代表了有社保有公积金  4 有社保无公积金  8 无社保有公积金 |
| job | 工作身份 | String |
| rqrDuration | 借款期限 | Int |
| rqrAmount | 借款额度 | Int |
| paymentDate | 用款时间 | Date |
| workPlace | 工作城市 | String |
| comment | 备注 | String |
| exeStatus | 客户进程 | int  1 有需求  2 有意向  3 已来访  4已签约  (5外包成交 6 会员成交) 都是成交 |
| customerCode | 客户编号 | String |
| mobile | 客户手机号码 | String |
| contactStatus | 拨打状态 | Int  0 销售未拨打客户  1：销售拨打客户，客户接通  2：销售拨打客户，客户未接通 |
| wechatFriendStatus | 微信好友关系状态 | Int  1 : 非微信好友  2：微信号不存在**(暂不用)**  3：微信好友申请中  4：微信好友申请已过期**(暂不用)**  5：拒绝添加好友**(暂不用)**  6：已经是微信好友 |

返回：

|  |
| --- |
| {  msg:"操作成功"  code：0  data：  {  total: 总条数  rows:[  {  第一个对象  },{  第二个 consumer对象  }  ]  }  } |

# 3: 编辑客户备注

## 2.1：请求方法和地址

接口方法：Post 请求地址： api/customer/editComment

## 2.2：请求参数

请求参数：【JSON格式】

|  |  |  |
| --- | --- | --- |
| Key | Value | 备注 |
| customerCode | 客户编号 | String |
| comment | 备注内容 | String |

请求案列：

|  |
| --- |
| http://10.40.4.110:8077/api/customer/editComment  {  "customerCode": "xxxxxxxxxxx",  "comment": "xxxxxxxxxxxx"  } |

## 2.3：响应结果

|  |
| --- |
| {  msg:"编辑备注成功"  code：0  data：null  } |

# 4: 获取短信模板

## 4.1：请求方法和地址

接口方法：Get 请求地址：api/sms/templates

## 4.2：请求参数

传入参数：无

## 4.3：响应结果

返回：SmsTemplate对象组成的list 【JSON格式】

SmsTemplate对象如下：

|  |  |  |
| --- | --- | --- |
| Key | Value | 备注 |
| id | 模板id | String |
| title | 模板标题 | String |
| content | 模板内容 | String |

返回样例：

|  |
| --- |
| {  msg:"操作成功"  code：0  data：  [  {  " id ": “0”,  " title ": "回访模板"  " content ": "恭喜你成功放款一个亿"  },  {  下一个对象  }  ]  } |

# 5: 同步短信到销售系统

## 5.1：请求方法和地址

接口方法：Post 请求地址：api/sms/addMessages

## 5.2：请求参数

传入参数：SmsMessage对象组成的List对象. 【JSON格式】

**对象必须是按照时间升序排序的。就是最后一条是最后的时间**

SmsMessage对象如下

|  |  |  |
| --- | --- | --- |
| Key | Value | 备注 |
| sender | 发送方编号 | String 销售工号或客户编号 |
| receiver | 接收方编号 | String 销售工号或客户编号 |
| senderRole | 谁发送 | int 1代表销售 2代表客户 |
| occurTime | 发送时间 | Date |
| content | 发送内容 | String |
| senderMobile | 发送者号码 | String |
| receiverMobile | 接受者号码 | String |
| senderName | 发送姓名 | String **注意格式** |
| receiverName | 接受姓名 | String |
| signature | 签名 | String  默认是【东方融资网】 |
| sendStatus | 发送状态 | **Int 成功是1 失败是0** |
| isRead | 读取状态 | Int 默认是1 |

注意：

1:在涉及到短信和电话的时候 编号就是工号 发送号码就是手机号码. 获取不到数据的时候就是空字符串。姓名也是如此。

**2:当发送者是销售的时候，**

**这个senderName =销售名字+[销售工号]组成，例如：张三[AA1234]**

**3:Signature 签名 默认是【东方融资网】**

3：sendStatus 发送状态 成功是1 失败是0

4：请求样例：

实际中获取不了客户的姓名和编号

|  |
| --- |
| http://10.40.4.110:8077/api/sms/addMessages  [{  "sender": "",  "receiver": "xxxxxxxx",  "senderRole": 2,  "occurTime": "2018-08-29 06:53:41.30",  "content": "xxxxxxxxx",  "senderMobile": "xxxxxxxxx",  "receiverMobile": "",  "senderName": "",  "receiverName": "xxxxxxx",  "signature": "【东方融资网】",  "sendStatus": 1,  "isRead": 1  },  {  "sender": "xxxxxxxxx",  "receiver": "",  "senderRole": 1,  "occurTime": "2018-08-29 06:53:41.30",  "content": "xxxxxxxxx",  "senderMobile": "",  "receiverMobile": "xxxxxxxxx",  "senderName": "xxxxxxxxxxxx",  "receiverName": "",  "signature": "【东方融资网】",  "sendStatus": 1,  "isRead": 1  }  ] |

## 5.3：响应结果

返回参数：

|  |
| --- |
| {  msg:"操作成功",  code:0,  **data:最后一次同步短信时间**  } |

# 6: 获取销售系统短信到手机

## 6.1：请求方法和地址

接口方法：Post 请求地址：api/sms/getMessages

## 6.2：请求参数

传入参数：【JSON格式】

|  |  |  |
| --- | --- | --- |
| Key | Value | 备注 |
| empCode | 销售工号 | String |
| customerCode | 客户编号 | String |
| empName | 销售姓名 | String |
| customerMobile | 客户手机号码 | String |
| pageSize | 每页记录行数 | Int |
| pageIndex | 初始化加载第几页 | Int |

注意：

1:在涉及到短信和电话的时候 编号就是工号 发送号码就是手机号码 获取不到数据的时候就是空字符串。

请求样例

|  |
| --- |
| http://10.40.3.110:8077/api/sms/getMessages  {   "empCode": "XXXX",   "customerCode": "XXXXX",   "customerMobile":"XXXXXXX",    "empName":"XXXXXXX",   "pageSize": 10,   "pageIndex": 1 } |

## 6.3：响应结果

返回：SmsMessage对象组成的list 【JSON格式】

|  |  |  |
| --- | --- | --- |
| Key | Value | 备注 |
| sender | 发送方编号 | String 销售工号或客户编号 |
| receiver | 接收方编号 | String 销售工号或客户编号 |
| senderRole | 谁发送 | int 1代表销售 2代表客户 |
| occurTime | 发送时间 | Date |
| content | 发送内容 | String |
| senderMobile | 发送者号码 | String |
| receiverMobile | 接受者号码 | String |
| senderName | 发送姓名 | String |
| receiverName | 接受姓名 | String |
| signature | 签名 | String 默认东方融资网 |
| sendStatus | 发送状态 | Int 默认是1 |
| isRead | 读取状态 | Int 默认是1 |

返回例子：

|  |
| --- |
| {  msg:"操作成功"  code：0  data：  {  total: 总条数  rows:[  {  第一个对象  },{  第二个 consumer对象  }  ]  }  } |

# 7: 同步通话记录到销售系统

## 7.1：请求方法和地址

接口方法：Post 请求地址：api/call/syncRecords

## 7.2：请求参数

传入参数 CallRecord对象组成的list 【JSON格式】

**对象必须是按照时间升序排序的。就是最后一条是最后的时间**

CallRecord对象如下：

|  |  |  |
| --- | --- | --- |
| Key | Value | 备注 |
| mobile | 通话号码 | String  永远是客户手机号码 |
| billSec | 通话时长(秒) | Int |
| callDate | 拨打时间 | Date |
| empWorkMobile | 销售手机号码 | String |
| empCode | 销售编号 | String |
| callStatus | 通话类型 | Int  0 销售未拨打客户  1：销售拨打客户，客户接通  2：销售拨打了客户，但是客户未接通  3：客户拨打销售，销售未接  4：客户拨打销售，销售接通了 |

根据通话类型

可以确认1：拨打者是什么角色 销售/客户

2：电话拨打状态 接听/呼叫/呼叫未接听

请求样例：(该样例只包含了一种情况，需要多种情况)

|  |
| --- |
| http://10.40.4.112:8077/api/call/syncRecords  [  {  "mobile":"xxxxxxxx",  "billSec":40,  "callDate":"2018-08-30 18:30:55",  "empWorkMobile":"",  "empCode":"XXXX",  "callStatus":1  }  ] |

## 7.3：响应结果

返回：

|  |
| --- |
| {  msg:"操作成功"  code:0  **data:最后一次通话记录同步时间**  } |

注意：

在涉及到短信和电话的时候，发送/接受都是销售的时候，那么发送编号就是工号 发送号码就是空字符串 是客户的时候，就一个是客户编号 一个是客户手机号码，后续存在工号和手机号码对应数据后能处理

# 8: 发送添加微信好友请求

## 8.1：请求方法和地址

接口方法：Post 请求地址：api/wechat/addFriend

## 8.2：请求参数

请求参数： 【JSON格式】

|  |  |  |
| --- | --- | --- |
| Key | Value | 备注 |
| customerMobile | 客户手机号码 | String 必须 |
| customerWechatId | 客户微信ID | String |
| **customerName** | **客户姓名，调用接口备注** | **String 必须** |
| empWorkMobile | 销售手机号码 | String 必须 |
| empWechatId | 销售微信ID | String 必须 |
| **empName** | **销售姓名 调用接口需要** | **String 必须** |

**注意：获取不到的值就是空字符串。必须字段除外**

请求示例：

|  |
| --- |
| {  "customerMobile": "18522222222",  "customerWechatId": "",  "customerName": "普陀",  "empWorkMobile": "18511111111",  "empWechatId": "wxid\_xbeakwi8vj9q22",  "empName": "黄埔"  } |

## 8.3：响应结果

|  |  |  |
| --- | --- | --- |
| Key | Value | 备注 |
| friendStatus | 微信好友关系状态 | Int  3：微信好友申请中 |

1：通过点击添加微信好友，传递客户手机号码。

2：后台调用接口，返回状态码

响应示例：

|  |
| --- |
| {  msg:"操作成功",  code：0,  data：:"3"  } |

# 9: 更新微信好友状态

## 9.1：请求方法和地址

接口方法：Post 请求地址：api/wechat/updateFriendStatus

## 9.2：请求参数

请求参数： 【JSON格式】

|  |  |  |
| --- | --- | --- |
| Key | Value | 备注 |
| customerMobile | 客户手机号码 | String 必须 |
| friendStatus | 微信好友关系状态 | Int 必须  1 : 非微信好友  6：已经是微信好友 |

请求示例：

|  |
| --- |
| {  "customerMobile": "18722222222",  "friendStatus": 1 // 或者6  } |

## 9.3：响应结果

无

响应示例：

|  |
| --- |
| {  msg:"操作成功",  code:0,  data:null  } |