# 软件售后服务系统接口规范文档

## 1. 服务人员

### 1.1 我的售后

1) URL地址: api/server

2) 请求方式: GET

3) 请求参数: {

serverName:xxx

}

4) 接口返回:

A:成功：{

Code:200,

Data:{

message: “success”,

list:[

{

id:xx,

username:xx,

softwareName:xx,

serverName:xx,

serviceState:xx

},

……

]

}

}

B:失败：　{

Code:500,

Date:{

message: “false”

}

}

### 1.2 售后提醒

1) URL地址: api/server

2) 请求方式: GET

3) 请求参数: {

serverName:xxx

}

4) 接口返回:

A:成功：{

Code:200,

Data:{

message: “success”,

list:[

{

id:xx,

sendName:xx,

messageDate:xx,

justMessage:xx,

},

……

]

}

}

B:失败：　{

Code:500,

Date:{

message: “false”

}

}

### 1.3 售后管理

1) URL地址: api/server

2) 请求方式: GET

3) 请求参数: {

serverName:xxx,

userName:xxx,

softwareName:xxx,

serviceKind:xxx

}

4) 接口返回:

A:成功：{

Code:200,

Data:{

message: “success”,

list:[

{

id:xx,

username:xx,

softwareName:xx,

serverName:xx,

serviceState:xx

},

……

]

}

}

B:失败：　{

Code:500,

Date:{

message: “false”

}

}

### 1.4 用户交互（界面）

1) URL地址: api/server

2) 请求方式: GET

3) 请求参数: {

sendName:xxx,

getName:xxx,

}

4) 接口返回:

A:成功：{

Code:200,

Data:{

message: “success”,

list:[

{

id:xx,

sendName:xx,

messageDate:xx,

justMessage:xx,

},

……

]

}

}

B:失败：　{

Code:500,

Date:{

message: “false”

}

}

### 1.4 用户交互（消息发送）

1) URL地址: api/server

2) 请求方式: GET

3) 请求参数: {

sendName:xxx,

getName:xxx,

justMessage:xxx,

messageDate:xxx

}

4) 接口返回:

A:成功：{

Code:200,

Data:{

message: “success”,

list:[

{

id:xx,

sendName:xx,

messageDate:xx,

justMessage:xx,

},

……

]

}

}

B:失败：　{

Code:500,

Date:{

message: “false”

}

}

## 2 FAQ搜索页面

### 2.1 显示FAQ

1) URL地址: api/FAQmessage

2) 请求方式: GET

3) 请求参数: {

showFAQ: “show”

}

4) 接口返回:

A:成功：{

Code:200,

Data:{

message: “success”,

list:[

{

id:xx,

faqName:xx,

faqDate:xx,

faqInfo:xx,

faqSoftware:xx,

},

……

]

}

}

B:失败：　{

Code:500,

Date:{

message: “false”

}

}

### 2.2 搜索FAQ

1) URL地址: api/FAQmessage

2) 请求方式: GET

3) 请求参数: {

searchFAQ:xxx

}

4) 接口返回:

A:成功：{

Code:200,

Data:{

message: “success”,

list:[

{

id:xx,

faqName:xx,

faqDate:xx,

faqInfo:xx,

faqSoftware:xx,

},

……

]

}

}

B:失败：　{

Code:500,

Date:{

message: “false”

}

}

## 3. 主页

### 3.1登陆

1) URL地址: api/login

2) 请求方式: GET

3) 请求参数: {

userName：“xxx”，

passWord：“xxx”，

Character：“xxx”

}

4) 接口返回:

A:成功：{

Code: 200,

Data:{

message：“success”

}

}

B:失败：　{

Code:500,

Data:{

message:“fail”

}

}

## 4. 客户

### 4.1 购买产品记录数据初始化

1. URL地址:api/client

2) 请求方式: GET

3) 请求参数:{

userName：“用户名称”

softwareName：“产品名称”，

buyDate：“购买日期”

}

1. 接口返回:

A：成功：{

Code：200，

Data：{

Message：“success”，

totalcount：“总共记录数”，

List：[

{

recordId：“购买记录id”，

softwareName：“软件名称”，

buyDate:“购买日期”，

}

......

]

}

}

B：失败：{

code:500，

Data{

Message：“fail”

}

}

### 4.2 售后查询模块

1. URL地址: api/client
2. 请求方式：get
3. 请求参数:{

UserId：“用户id”，

ServeId：“服务编号”

serviceKind：“问题类型”，

serviceInfo：“问题描述”，

serviceState：“维护状态”，

serverName：“服务人员”，

softwareName：“软件名称”，

}

1. 接口返回：

A：成功：{

Code：200，

Data：{

Message：“success”，

List[

{

UserId：“用户id”，

ServeId：“服务编号”

serviceKind：“问题类型”，

serviceInfo：“问题描述”，

serviceState：“维护状态”，

serverName：“服务人员”，

softwareName：“软件名称”，

}

.......

]

}

}

B:失败：{

Code：500，

Data{

Message：“fail”

}

}

### 4.3售后服务申请功能

1）url：api/client

2）请求方式：post

3）请求参数：{

UserId：“用户id”，

ServeID：“服务编号”

serviceKind：“问题类型”，

serviceInfo：“问题描述”，

serviceState：“维护状态”，

serverName：“服务人员”，

softwareName：“软件名称”，

}

1. 接口返回：

A:成功：{

Code: 200,

Data:{

message：“success”

}

}

B:失败：　{

Code:500,

Data:{

message:“fail”

}

}

### 4.4 撤销服务功能

1）url：api/client

2）请求方式：put

3）请求参数：{

ServeId：“服务编号”

}

4）接口返回：

A:成功：{

Code: 200,

Data:{

message：“success”

}

}

B:失败：　{

Code:500,

Data:{

message:“fail”

}

}

### 4.5修改服务功能

1. url：api/client
2. 请求方式：put

3）请求参数：{

UserId：“用户id”，

ServeID：“服务编号”

serviceKind：“问题类型”，

serviceInfo：“问题描述”，

serviceState：“维护状态”，

serverName：“服务人员”，

softwareName：“软件名称”，

}

4）接口返回：

A：成功：{

Code：200，

Data：{

Message：“success”，

List[

{

UserId：“用户id”，

ServeId：“服务编号”

serviceKind：“问题类型”，

serviceInfo：“问题描述”，

serviceState：“维护状态”，

serverName：“服务人员”，

softwareName：“软件名称”，

}

.......

]

}

}

B:失败：{

Code：500，

Data{

Message：“fail”

}

}

### 4.6显示已完成服务功能

1）url：api/client

2）请求方式：GET

3）请求参数：{

UserId：“用户id”，

ServeID：“服务编号”

serviceKind：“问题类型”，

serviceInfo：“问题描述”，

serviceState：“维护状态”，

serverName：“服务人员”，

softwareName：“软件名称”，

}

4）接口返回：

A：成功：{

Code：200，

Data：{

Message：“success”，

List[

{

UserId：“用户id”，

ServeId：“服务编号”

serviceKind：“问题类型”，

serviceInfo：“问题描述”，

serviceState：“维护状态”，

serverName：“服务人员”，

softwareName：“软件名称”，

}

.......

]

}

}

B:失败：{

Code：500，

Data{

Message：“fail”

}

}

### 4.7 显示未完成服务功能

1）url：api/client

2）请求方式：GET

3）请求参数：{

UserId：“用户id”，

ServeID：“服务编号”

serviceKind：“问题类型”，

serviceInfo：“问题描述”，

serviceState：“维护状态”，

serverName：“服务人员”，

softwareName：“软件名称”，

}

4）接口返回：

A：成功：{

Code：200，

Data：{

Message：“success”，

List[

{

UserId：“用户id”，

ServeId：“服务编号”

serviceKind：“问题类型”，

serviceInfo：“问题描述”，

serviceState：“维护状态”，

serverName：“服务人员”，

softwareName：“软件名称”，

}

.......

]

}

}

B:失败：{

Code：500，

Data{

Message：“fail”

}

}

## 5. 超级管理员

### 5.1 向服务器获取异常售后服务的服务列表

1) URL地址: /api/getAbnormalServiceInf

2) 请求方式: GET

3) 请求参数:

{

pageNo：“x,

ageSize:"8"

}

接口返回:

A:成功：{

Code: 200,

Data:{

message: "success",

pageNo: x,

pageSize: x,

totalPage: x,

list: [

{

id: xxx,

applicant: "xxx"，

serviceState：“xxx”,

reason:"xxx"

},

{

id: xxx,

applicant: "xxx"，

serviceState：“xxx”,

reason:"xxx"

},

........

],

}

}

B:失败：　{

Code:500,

Data:{

message:“fail”，

}

}

### 5.2 向服务器提交异常售后服务处理信息

URL地址: /api/handlingExceptionServices

1. 请求方式: GET
2. 请求参数: {

id: XXX,

mode:"XXX"

}

4)接口返回:

A:成功：{

Code: 200,

Data:{message: "success",

}

}

B:失败：　{

Code:500,

Data:{message:“fail”}

}

### 5.3 向服务器获取权限设置列表

URL地址: /api/GetPermissionSettingsList

请求方式: GET

3) 请求参数: {

pageNo: x,

pageSize: x,

}

4）接口返回:

A:成功：{

Code: 200,

Data:{

message: "success",

list:[

{

id:"xxx",

roal:"xxx",

userState:"xxx"

},

......

{

id:"xxx",

role:"xxx",

userState:"xxx"

},

....

]

}

}

B:失败：　{

Code:500,

Data:{

message:“fail”

}

}

### 5.4 向服务器提交用户权限设置请求

1)URL地址: /api/permissionSettingRequest

2)请求方式: POST

3)请求参数: {

id:"xxx",

role:"xxx",

solution：”xxx”

}

4）接口返回:

A:成功：{

Code: 200,

Data:{

message: "success",

}

}

B:失败：　{

Code:500,

Data:{

message:“fail”

}

}

### 5.5 向服务器获取faq列表

1)URL地址: /api/getFaqList

2)请求方式: get

3)请求参数: {

showFaq: “show”

}

4）接口返回:

A:成功：{

Code: 200,

Data:{

message: "success",

totalPage: “xxx”

List:[

{

faqName:”xxx”,

Faqinfo:”xxx”,

faqDate:”xxx”

},

...

{

faqName:”xxx”,

Faqinfo:”xxx”,

faqDate:”xxx”

}]

}

}

B:失败：　{

Code:500,

Data:{

message:“fail”

}

}

### 5.6.向服务器提交faq修改

1)URL地址: /api/FaqModify

2)请求方式: POST

3)请求参数: {

faqName:”xxx”,

Faqinfo:”xxx”,

}

4）接口返回:

A:成功：{

Code: 200,

Data:{

message: "success",

}

}

B:失败：　{

Code:500,

Data:{

message:“fail”

}

}

### 5.7 向服务器提交faq删除

1)URL地址: /api/FaqDelete

2)请求方式: POST

3)请求参数: {

faqName:”xxx”,

Faqinfo:”xxx”,

}

4）接口返回:

A:成功：{

Code: 200,

Data:{

message: "success",

}

}

B:失败：　{

Code:500,

Data:{

message:“fail”

}

}

### 5.8 向服务器提交faq新增请求

1)URL地址: /api/Faqadd

2)请求方式: POST

3)请求参数: {

faqName:”xxx”,

faqinfo:“xxx”,

faqSoftware:”xxx”,

type:”xxx”

author:”xxx”,

solution:”xx”

}

4）接口返回:

A:成功：{

Code: 200,

Data:{

message: "success",

}

}

B:失败：　{

Code:500,

Data:{

message:“fail”

}

}

### 5.9 向服务器提交faq搜索请求

1) URL地址: api/FAQmessage

2) 请求方式: GET

3) 请求参数: {

searchFAQ:xxx

}

4) 接口返回:

A:成功：{

Code:200,

Data:{

message: “success”,

list:[

{

id:xx,

faqName:xx,

faqDate:xx,

faqInfo:xx,

faqSoftware:xx,

},

……

]

}

}

B:失败：　{

Code:500,

Date:{

message: “false”

}

}

B:失败：　{Code:500,Data:{message:“fail”}}

### 5.10 更新版本信息

1)URL地址: /api/GetVersionInformation

2)请求方式: GET

3)请求参数: {

softwareName:xxx,

softwareInfo:xxx

}

4）接口返回:

A:成功：{

Code: 200,

Data:{

message: "success",

List:[

{

id:xxx,

softwareName:xxx,

softwareInfo:xxx,

updateDate:xxx

},...{

id:xxx,

softwareName:xxx,

softwareInfo:xxx,

updateDate:xxx

}]

}

}

B:失败：　{

Code:500,

Data:{

message:“fail”

}

}

## 6. 参考信息

code : 请求处理状态

200: 请求处理成功

500: 请求处理失败

401: 请求未认证，跳转登录页

406: 请求未授权，跳转未授权提示页

data.message: 请求处理消息

code=200 且 data.message="success": 请求处理成功

code=200 且 data.message!="success": 请求处理成功, 普通消息提示：message内容

code=500: 请求处理失败，警告消息提示：message内容