



汉朔科技PriSmart 1.3.2 API使用说明书

V1.0.1

HS-PRISMART-V132005

郑重声明

本文档以及其所包含的内容为汉朔科技股份有限公司所有，并受中国法律和其他可适用的国际公约的版权保护。未经汉朔的事先书面授权，不得对本文档的全部或者部分进行任何形式的复制、传输、披露、修订、修改或使用。违反者将对其违反行为所造成的任何以及全部损害承担责任，汉朔保留采取任何法律所允许范围内救济措施的权利。

关于本文档

感谢使用汉朔科技 PriSmart 1.3.2 智能零售系统。

如果您第一次使用该产品，我们建议您一定要先阅读这本文档，并且理解文档里提及的各项内容。如果您遇到难以确定或难以解决的难题，请您直接联系汉朔科技客户服务中心（中国：400-0365-305；荷兰：0800-022-5037；比利时：0800-71-335；法国：0800-91-7602；泰国：1800-011-185），我们竭诚为您提供多渠道技术服务。

适用对象

本文档介绍了 PriSmart 接口使用、包含的基本功能、性能特点和使用注意事项等。

系统接入 API 面向具有一定的软件开发能力，了解 WEB 软件开发、JAVA 语言、JSON 的软件开发人员、研发管理人员。本文档适用于以下人员使用阅读：

- 手持设备开发人员
- JAVA 软件工程师
- 其他语言软件工程师

符号说明

图标	说明
⚠	本图标标识的信息需要读者特别注意，引起读者高度重视
📖	本图标标识的信息是对正文内容的解释说明，为读者更好的理解文档内容
[X-X]	表示此处有名词的详细解释说明

术语解释

术语	全称	解释
AP	Wireless Access Point	价签控制器（ESL Controller），也称为基站
API	Application Programming Interface	应用程序编程接口，是一些预先定义的函数，目的是提供应用程序与开发人员基于某软件或硬件得以访问

术语	全称	解释
		一组例程的能力，而又无需访问源码，或理解内部工作机制的细节。
ESL	Electronic Shelf Label	电子价签
ESL-Working	价签管理系统	管理门店内的所有基站和价签设备，3.0 之后版本可支持多门店。
PDA	Personal Digital Assistant	智能手持终端，手持终端对门店内价签具体操作。
PriSmart	PriSmart 智能零售系统	管理所有门店商品数据并下发价签变价，支持多门店管理。
XML-RPC	XML Remote Procedure Call	XML（标准通用标记语言下的一个子集）远程过程调用，是一套允许运行在不同操作系统、不同环境的程序实现基于 Internet 过程调用的规范和一系列的实现。这种远程过程调用使用 http 作为传输协议，XML 作为传送信息的编码格式。XML-RPC 的定义尽可能的保持了简单，但同时能够传送、处理、返回复杂的数据结构。

目录

1 接入前说明	1
2 接入说明	2
2.1 接入方式	2
2.2 身份认证	2
2.3 返回参数说明	3
2.4 版本兼容性	3
3 接口描述	4
3.1 login POST 门店用户登录	4
3.1.1 API 说明	4
3.1.2 请求参数	4
3.1.3 返回参数	4
3.1.4 示例	5
3.2 changePwd POST 修改密码	6
3.2.1 API 说明	6
3.2.2 请求参数	6
3.2.3 返回参数	6
3.2.4 示例	6
3.3 logout POST 登出系统	7
3.3.1 API 说明	7
3.3.2 请求参数	7
3.3.3 返回参数	8
3.3.4 示例	8
3.4 getGoodsByIdOrBarcode POST 按 EAN 或 SKU 查询商品	8
3.4.1 API 说明	8
3.4.2 请求参数	8
3.4.3 返回参数	9
3.4.4 示例	13
3.5 bind POST 建立价签和商品的绑定关系	14
3.5.1 API 说明	14
3.5.2 请求参数	14
3.5.3 返回参数	15
3.5.4 示例	15
3.6 unbind POST 解除价签和商品的绑定关系	17

3.6.1 API 说明	17
3.6.2 请求参数.....	17
3.6.3 返回参数.....	17
3.6.4 示例.....	18
3.7 setEslControl POST 控制价签闪灯和切页.....	19
3.7.1 API 说明	19
3.7.2 请求参数.....	19
3.7.3 返回参数.....	20
3.7.4 示例.....	20
3.8 getPr POST 查询绑定关系.....	23
3.8.1 API 说明	23
3.8.2 请求参数.....	23
3.8.3 返回参数.....	23
3.8.4 示例.....	24
3.9 integration POST 数据对接.....	26
3.9.1 API 说明	26
3.9.2 请求参数.....	26
3.9.3 返回参数.....	28
3.9.4 示例.....	29
3.10 integration/flash POST 控制门店全部价签闪灯/灭灯.....	33
3.10.1 API 说明	33
3.10.2 请求参数.....	33
3.10.3 返回参数.....	33
3.10.4 示例.....	33

1 接入前说明

系统接入开发时，除了需要满足每个接口的规范限制、调用频率限制外，还需特别注意消息、数据等敏感信息的使用。请严格按照规范要求调用。在使用前请进行充分测试。

涉及用户数据时：

- 收集传输可用的数据后，必须采取必要的保护措施，防止数据被盗、泄漏等。
- 一旦您停止使用本服务，或我们基于任何原因终止您使用本服务，请停止访问及数据接入。

2 接入说明

2.1 接入方式

如下 API 接口采用 HTTP 协议请求与响应，请求内容以 JSON 作为数据体。

关于接入方式，您将通过 HTTP 协议 POST 请求方式发送 JSON 数据到 PriSmart 应用服务器，当接收到数据并验证无误后返回消息，返回数据告知您此次提交是否成功。

2.2 身份认证

在本文档中，登录（login）接口、对接（integration）接口和控制门店全部价签闪灯（integration/flash）接口无需认证可直接调用，其余接口需认证后方可调用。

认证需要调用登录（login）接口获取 jsessionid，其他需要认证的接口的调用均要带上 jsessionid，详见下方各接口详细说明。

使用 web 页面调用登录接口获取 jsessionid 后，在后续的调用中，浏览器默认会自动将 jsessionid 参数存入 cookie 并在请求中发送，无需特殊处理。在无法使用 cookie 的场景中，可通过如下两种方式提供 jsessionid：

- 在 http 请求标头中添加 Cookie 参数，并将 jsessionid 的值按格式“jsessionid=xxx”的方式赋值给 Cookie 参数，例如：

```
Accept: application/json, text/plain, */*
Accept-Encoding: gzip, deflate, br
Accept-Language: zh-CN, zh; q=0.9
Connection: keep-alive
Content-Type: application/json; charset=utf-8
Cookie: JSESSIONID= 535537975674130432_ofmsrlflb0gj129jwr79xkqz64
```

- 将 jsessionid 参数使用 “;” 拼接在接口 url 后，例如：

```
http://localhost:8080/prismart/logout;jsessionid=xxx
```

说明：下文需要身份认证的接口描述中，均采用第一种方式，即在 http 请求标头中添加 Cookie 参数，并将 jsessionid 的值按格式“jsessionid=xxx”的方式赋值给 Cookie 参数。

2.3 返回参数说明

变量名	描述
resultCode	<p>状态码。</p> <ul style="list-style-type: none"> ● 1001: 成功 ● 1002: 失败 ● 1003: 请求参数错误 ● 1004: 权限异常 ● 1005: 商品不存在 ● 1006: 价签不存在 ● 1007: 会话失效 ● 1008: 用户不存在 ● 1009: 指定的资源不存在 ● 1010: EAN 码对应多个商品，无法通过 EAN 确定商品 ● 1011: 创建的资源与现有数据冲突 ● 1101: 密码错误 ● 1102: 用户状态异常
result	<p>返回结果。</p> <p>提示消息仅供参考，可根据状态码自行处理状态或操作提示。</p>
errorContext	<p>关键内容。</p> <p>提示消息仅供参考，一般为错误的关键字段，如 password。</p>
message	<p>返回信息。</p> <p>不同返回体的 message 内容值不同。一般为具体描述，如 password is null。</p>
data	<p>数据内容。</p> <p>不同返回体的 data 内容值不同。详细 data 信息参见每个接口描述中的返回参数体章节。</p>

 **说明：**为兼容早期的版本，有些返回格式会以老版格式返回，不影响当前使用。

2.4 版本兼容性

1. 兼容以前版本 PriSmart 的 API。
2. 除对接接口（integration）外，兼容 ShopWeb 的 API。
3. 对接接口（integration）在 ShopWeb 中返回成功的 resultCode 状态码为 0，而在 PriSmart 中为 1001，在 ShopWeb 迁移 PriSmart 时需要注意。

3 接口描述

3.1 login POST 门店用户登录

3.1.1 API 说明

URL地址	http://{PriSmart访问地址}/login
METHOD	POST
内容类型	application/json;charset=utf-8
功能描述	通过门店账号的用户名和密码登录PriSmart系统，仅能管理该门店，当门店账号下管理1个以上门店时无法使用该接口登录。
需要身份认证	否
示例	http://192.168.1.100:8080/prismart/login

3.1.2 请求参数

Map 类型，详细参数如下：

- **username:** String。必填。登录用户名。
- **password:** String。必填。登录密码。

3.1.3 返回参数

- **resultCode:** Integer。状态码。参考 [2.3 返回参数说明](#)。
- **result:** String。返回结果。
- **message:** String。返回消息。
- **data:** Map。返回数据内容。
 - **jsessionid:** String。身份标识。
 - **admin:** Integer。是否为管理员。0：非管理员；1：管理员。
 - **customerCode:** String。客户号。
 - **storeName:** String。门店名称。
 - **storeCode:** String。门店编号。

3.1.4 示例

3.1.4.1 请求示例

```
POST http://192.168.1.100:8080/prismart/login
{
    "username": "customer01.store01",
    "password": "000000"
}
```

3.1.4.2 返回示例

3.1.4.2.1 正常返回示例

```
{
    "resultCode": 1001,
    "result": "succeed",
    "message": "success",
    "data": {
        "jsessionid": "342714307704238080_1u3akp3plbmzmh87j5qiqmj060",
        "admin": 0,
        "storeName": "storeName01",
        "storeCode": "storeCode01",
        "customerStoreCode": "customerStoreCode01"
    }
}
```

3.1.4.2.2 异常返回示例

```
{
    "resultCode": 1101,
    "result": "password_mismatch",
    "message": "Incorrect password of user \"customer01.store01\""
}
```

3.2 changePwd POST 修改密码

3.2.1 API 说明

URL地址	http://{PriSmart访问地址}/changePwd
METHOD	POST
内容类型	application/json; charset=utf-8
功能描述	修改指定用户的密码。
需要身份认证	是。参考 2.2 身份认证 。
示例	http://192.168.1.100:8080/prismart/changePwd

3.2.2 请求参数

Map 类型，详细参数如下：

- **username:** String。必填。用户名。
- **oldPassword:** String。必填。旧密码。
- **password:** String。必填。新密码。
- **customerCode:** String。选填。客户号。
- **storeCode:** String。选填。门店号。

3.2.3 返回参数

- **resultCode:** Integer。状态码。参考[2.3 返回参数说明](#)。
- **result:** String。返回结果。
- **message:** String。返回消息。
- **errorContext:** String。错误信息中的关键内容。

3.2.4 示例

```
{
    "customerCode": "customer01",
    "storeCode": "store01",
```

```

    "username": "customer01.store01",
    "password": "000000",
    "oldPassword": "000000"
}

```

3.2.4.2 返回示例

3.2.4.2.1 正常返回示例

```

{
    "resultCode": 1001,
    "result": "succeed",
    "message": "success"
}

```

3.2.4.2.2 异常返回示例

```

{
    "resultCode": 1003,
    "result": "illegal_arguments",
    "errorContext": "oldPassword",
    "message": "oldPassword is null"
}

```

3.3 logout POST 登出系统

3.3.1 API 说明

URL地址	http://{PriSmart访问地址}/logout
METHOD	POST
内容类型	application/json;charset=utf-8
功能描述	退出登录。
需要身份认证	是。参考 2.2 身份认证 。
示例	http://192.168.1.100:8080/prismart/logout

3.3.2 请求参数

无。

3.3.3 返回参数

- **resultCode:** Integer。状态码。参考 [2.3 返回参数说明](#)。
- **result:** String。返回结果。
- **message:** String。返回消息。

3.3.4 示例

3.3.4.1 请求示例

```
POST http://127.0.0.1:8080/prismart/logout
```

3.3.4.2 返回示例

```
{
    "resultCode": 1001,
    "result": "succeed",
    "message": "success"
}
```

3.4 getGoodsByIdOrBarcode POST 按 EAN 或 SKU 查询商品

3.4.1 API 说明

URL地址	http://{PriSmart访问地址}/getGoodsByIdOrBarcode
METHOD	POST
内容类型	application/json;charset=utf-8
功能描述	按EAN或SKU查询商品信息。
需要身份认证	是。参考 2.2身份认证 。
示例	http://192.168.1.100:8080/prismart/getGoodsByIdOrBarcode

3.4.2 请求参数

Map 类型，详细参数如下：

- **code:** String。必填。商品 SKU 或 EAN。

3.4.3 返回参数

- **sid:** String。返回空字符串。
- **resultCode:** Integer。状态码。参考 [2.3 返回参数说明](#)。
- **result:** String。返回结果。
- **message:** String。返回消息。
- **data:** Map。商品数据内容。
 - **id:** String。商品 ID。格式为：客户号/门店号/商品 SKU，例如：customer01/store01/1001。
 - **sku:** String。商品 SKU。
 - **customerStoreCode:** String。客户号。
 - **storeCode:** String。门店号。
 - **itemName:** String。商品名称。
 - **price1:** BigDecimal。零售价。
 - **itemShortName:** String。商品简称。
 - **freshItemCode:** String。生鲜商品码。
 - **ean:** String。商品条码。
 - **itemStatus:** String。商品状态。
 - **brand:** String。商品品牌。
 - **unit:** String。商品单位。
 - **saleMode:** String。销售方式。
 - **placeOfOrigin:** String。商品产地。
 - **specification:** String。商品规格。
 - **qrCode:** String。商品二维码。
 - **packSize:** Integer。商品箱规。
 - **grade:** String。商品等级。
 - **description:** String。商品描述/备注。
 - **level1CategoryCode:** String。一级分类编号。

- **level1CategoryName:** String。一级分类名称。
- **level2CategoryCode:** String。二级分类编号。
- **level2CategoryName:** String。二级分类名称。
- **level3CategoryCode:** String。三级分类编号。
- **level3CategoryName:** String。三级分类名称。
- **level4CategoryCode:** String。四级分类编号。
- **level4CategoryName:** String。四级分类名称。
- **level5CategoryCode:** String。五级分类编号。
- **level5CategoryName:** String。五级分类名称。
- **supevisedBy:** String。监制单位。
- **supervisionHotline:** String。监督电话。
- **pricingStaff:** String。计价员。
- **Price1Description:** String。零售价描述。
- **price2:** BigDecimal。促销价。
- **price2Description:** String。促销价描述。
- **price3:** BigDecimal。会员价。
- **price3Description:** String。会员价描述。
- **price4:** BigDecimal。其他价格。
- **price4Description:** String。其他价格描述。
- **price5:** BigDecimal。市场零售价。
- **price5Description:** String。市场零售价描述。
- **price6:** BigDecimal。折算价。
- **price6Description:** String。折算价格描述。
- **promoDateFrom:** Date。促销开始日期。
- **promoDateTo:** Date。促销结束日期。
- **promoTimeFrom:** Time。促销开始时间。
- **promoTimeTo:** Time。促销结束时间。

- **vipDateFrom:** Date。会员促销开始日期。
- **vipDateTo:** Date。会员促销结束日期。
- **vipTimeFrom:** Time。会员促销开始时间。
- **vipTimeTo:** Time。会员促销结束时间。
- **promoReason:** String。促销原因。
- **promoFlag:** Integer。促销标识。
- **inventory:** BigDecimal 。库存。
- **safetyStock:** BigDecimal。安全库存。
- **dms:** BigDecimal。日均销售数量。
- **todaySalesQty:** BigDecimal。当日销售数量。
- **cumulatedSalesQty:** BigDecimal。累计销售数量。
- **shelfQty:** BigDecimal。陈列量。
- **shelfCode:** String 。货架编号/货位号。
- **shelfTier:** String。货架层。
- **shelfColumn:** String。货架列。
- **displayLocation:** String。摆放位置。
- **supplierCode:** String。供应商编号。
- **supplierName:** String。供应商名称。
- **onOrderQty:** BigDecimal。在途订单数量。
- **expiryDate:** Date。过期日。
- **storageLife:** String。保质期限。
- **shelfLife:** Integer。已上架时间。
- **ingredientTable:** String。配料表。
- **manufacturer:** String。生产商名称。
- **manufacturerAddress:** String。生产商地址。
- **memberPoints:** BigDecimal。会员积分。
- **rsrvTxt1:** String。保留字段 1_文本。

- **rsrvTxt2:** String。保留字段 2_文本。
- **rsrvTxt3:** String。保留字段 3_文本。
- **rsrvTxt4:** String。保留字段 4_文本。
- **rsrvTxt5:** String。保留字段 5_文本。
- **rsrvTxt6:** String。保留字段 6_文本。
- **rsrvTxt7:** String。保留字段 7_文本。
- **rsrvTxt8:** String。保留字段 8_文本。
- **rsrvTxt9:** String。保留字段 9_文本。
- **rsrvTxt10:** String。保留字段 10_文本。
- **rsrvDec1:** BigDecimal。保留字段 1_小数。
- **rsrvDec2:** BigDecimal。保留字段 2_小数。
- **rsrvDec3:** BigDecimal。保留字段 3_小数。
- **rsrvDec4:** BigDecimal。保留字段 4_小数。
- **rsrvDec5:** BigDecimal。保留字段 5_小数。
- **RsrvInt1:** Integer。保留字段 1_整数。
- **RsrvInt2:** Integer。保留字段 2_整数。
- **RsrvInt3:** Integer。保留字段 3_整数。
- **RsrvInt4:** Integer。保留字段 4_整数。
- **RsrvInt5:** Integer。保留字段 5_整数。
- **RsrvBlob:** String。保留字段 1_长文本。
- **template:** String。模板名称。
- **updateCount:** Integer。已更新次数。
- **status:** Integer。状态。
- **createDate:** Date。创建时间。
- **updateDate:** Date。更新时间。

3.4.4 示例

3.4.4.1 请求示例

```
POST http://192.168.1.100:8080/prismart/getGoodsByIdOrBarcode
{
    "code": "001"
}
```

3.4.4.2 返回示例

3.4.4.2.1 正常返回示例

```
{
    "sid": "",
    "message": "success",
    "result": "success",
    "resultCode": 1001,
    "data": [
        {
            "sku": "001",
            "itemName": "apple",
            "price1": 10.00,
            ...
        },
        {
            "sku": "002",
            "itemName": "banana",
            "price1": 10.00,
        }
    ],
    "promoDateFrom" : 1642780800000,
    "promoDateTo" : 1642867200000,
    "promoFlag" : 0,
    "lastBindTime" : 1641552039151,
    "updateCount" : 4,
    ...
}
]
```

3.4.4.2.2 异常返回示例

```
{
    "sid": "",
    "message": "param error",
    "result": "fail",
    "resultCode": 1003
}
```

3.5 bind POST 建立价签和商品的绑定关系

3.5.1 API 说明

URL地址	http://{PriSmart访问地址}/bind
METHOD	POST
内容类型	application/json; charset=utf-8
功能描述	根据价签ID、商品SKU或ID，将价签与商品进行绑定。
需要身份认证	是。参考 2.2 身份认证 。
示例	http://192.168.1.100:8080/prismart/bind

3.5.2 请求参数

List 类型，详细参数如下：

- **esId: String**。必填。价签标识。
- **goodsId: String**。与 **goodsSku** 参数必填其一。两个参数都填时，此参数优先级高。商品 ID。格式为：客户号/门店号/商品 SKU，例如：customer01/store01/1001。
- **goodsSku: String**。与 **goodsId** 参数必填其一。两个参数都填时，**goodsId** 优先级高。商品 SKU。
- **position: Integer**。选填。分屏显示时当前商品要绑定的位置。
- **extra: Map**。选填。扩展参数。

3.5.3 返回参数

- **resultCode:** Integer。状态码。参考 [2.3 返回参数说明](#)。
- **result:** String。返回结果。
- **message:** String。返回消息。
- **data:** Map list。返回数据内容。
 - **esId:** String。价签标识。
 - **resultCode:** Integer。状态码。参考 [2.3 返回参数说明](#)。
 - **result:** String。返回结果。
 - **message:** String。返回消息。

3.5.4 示例

3.5.4.1 请求示例

```
POST http://192.168.1.100:8080/prismart/bind
[
  {
    "esId": "50-1A-4C-EE",
    "goodsId": "##/xxx/2",
    "position": 2,
    "extra": {
      "faceX": "001",
      "faceY": "002",
      ...
    }
  },
  {
    "esId": "50-1A-4C-EF",
    "goodsSku": "001",
    "extra": {
      "faceZ": "003",
      "faceY": "002",
      ...
    }
  }
]
```

3.5.4.2 返回示例

3.5.4.2.1 正常返回示例

```
{  
    "resultCode": 1001,  
    "result": "succeed",  
    "message": "success",  
    "data": [  
        {  
            "eslId": "50-1A-4C-EE",  
            "message": "success",  
            "result": "succeed",  
            "resultCode": 1001  
        },  
        {  
            "eslId": "50-1A-4C-EF",  
            "message": "success",  
            "result": "succeed",  
            "resultCode": 1001  
        }  
    ]  
}
```

3.5.4.2.2 异常返回示例

```
{  
    "resultCode": 1001,  
    "result": "succeed",  
    "message": "success",  
    "data": [  
        {  
            "eslId": "10-20-35-50",  
            "message": "Specified ESL ID is not existed",  
            "result": "nonexistent_esl",  
            "resultCode": 1006,  
            "errorContext": "eslId"  
        }  
    ]  
}
```

3.6 unbind POST 解除价签和商品的绑定关系

3.6.1 API 说明

URL地址	http://{PriSmart访问地址}/unbind
METHOD	POST
内容类型	application/json;charset=utf-8
功能描述	使用价签标识对价签进行解绑，若不指定goodsId，会解绑该价签上所有的商品；若指定了goodsId，则只会将价签上对应SKU的商品解绑。
需要身份认证	是。参考 2.2 身份认证 。
示例	http://192.168.1.100:8080/prismart/unbind

3.6.2 请求参数

List 类型，详细参数如下：

- **esId:** String。必填。价签标识。
- **goodsId:** String。必填。格式为：客户号/门店号/商品 SKU，商品 ID。例如 customer01/store01/1001。

3.6.3 返回参数

- **resultCode:** Integer。状态码。参考[2.3 返回参数说明](#)。
- **result:** String。返回结果。
- **message:** String。返回消息。
- **data:** Map list。返回数据内容。
 - **esId:** String。价签标识。
 - **resultCode:** Integer。状态码。参考[2.3 返回参数说明](#)。
 - **result:** String。返回结果。
 - **message:** String。返回消息。

3.6.4 示例

3.6.4.1 请求示例

```
POST http://192.168.1.100:8080/prismart/unbind
[
  {
    "eslId": "50-1A-4C-EE"
  },
  {
    "goodsId": "customer01/store01/12",
    "eslId": "55-2E-18-1C"
  }
]
```

3.6.4.2 回执示例

3.6.4.2.1 正常返回示例

```
{
  "resultCode": 1001,
  "result": "succeed",
  "message": "success",
  "data": [
    {
      "eslId": "50-1A-4C-EE",
      "message": "success",
      "result": "succeed",
      "resultCode": 1001
    }
  ]
}
```

3.6.4.2.2 异常返回示例

```
{
  "resultCode": 1001,
  "result": "succeed",
  "message": "success",
  "data": [
    {

```

```

        "eslId": "50-1A-4C-EE",
        "message": "Specified ESL ID \"50-1A-4C-EE\" is not existed",
        "result": "nonexistent_esl",
        "resultCode": 1006,
        "errorContext": "eslId"
    }
]
}

```

3.7 setEsControl POST 控制价签闪灯和切页

3.7.1 API 说明

URL地址	http://{PriSmart访问地址}/setEsControl
METHOD	POST
内容类型	application/json;charset=utf-8
功能描述	根据价签标识或商品SKU，控制价签闪灯切页行为。如传递led_color、led_count参数，但page_id、page_time参数没有传递，则进行亮灯控制；如传递page_id、page_time参数，但led_color、led_count参数没有传递，则进行切页控制；如四个参数都传递则同时亮灯及切页。
需要身份认证	是。参考 2.2 身份认证 。
示例	http://192.168.1.100:8080/prismart/setEsControl

说明：接口存在参数组合规则，规则如下：

- 此接口仅支持控制一组商品或价签。
- 需配合 ESL-Working 2.5 及以上版本使用。

3.7.2 请求参数

List 类型。详细参数如下：

- **eslId: String**。必填。价签标识。
- **goodsId: String**。商品 ID。与 **goodsSku** 参数必填其一。两个参数都填时，此参数优先级高，忽略 **goodsSku** 参数。格式为：客户号/门店号/商品 SKU，例如 customer01/store01/12。
- **goodsSku: String**。商品 SKU。与 **goodsId** 参数必填其一。两个参数都填时，**goodsId** 优先级高，忽略此参数。

- **led_color:** String 数组。选填。闪灯颜色。传递多种颜色时会按照颜色依次展示。当前支持颜色值包括: blue、red、violet、green、indigo、yellow、white。
- **led_count:** String。选填。闪灯循环次数。取值范围 0-65535。值为 0 时，表示灭灯。
- **led_on_time:** String。选填。灯亮时长。取值范围 0-255，单位 30ms。
- **led_off_time:** String。选填。灯灭时长。取值范围 0-255，单位 30ms。
- **led_sleep_time:** String。选填。两次循环间隔时间。取值范围 0-65535，单位 30ms。
- **page_id:** String。选填。切页时，目标页 ID。取值范围 0-7。
- **page_time:** String。选填。在目标页的停留时间。取值范围 0-65535。单位 s。结束后会切换回模板中的默认页。取值为 0 时，表示不执行切页动作。

3.7.3 返回参数

- **resultCode:** Integer。状态码。参考 [2.3 返回参数说明](#)。
- **result:** String。返回结果。
- **message:** String。返回消息。
- **data:** Map list。返回数据内容。
 - **esId:** String。价签标识。
 - **resultCode:** Integer。状态码。参考 [2.3 返回参数说明](#)。
 - **result:** String。返回结果。
 - **message:** String。返回消息。

3.7.4 示例

3.7.4.1 请求示例

3.7.4.1.1 闪灯请求示例

```
POST http://192.168.1.100:8080/prismart/setEslControl
[
  {
    "esId": "59-12-10-66",
    "led_color": ["red"],
```

```
    "led_count": "30"
  },
{
  "eslId": "53-8F-F8-14",
  "led_color": ["blue"],
  "led_count": "10",
  "led_on_time": "10",
  "led_off_time": "30",
  "led_sleep_time": "40"
}
]
```

3.7.4.1.2 灭灯请求示例

```
POST http://192.168.1.100:8080/prismart/setEslControl
[
  {
    "eslId": "59-12-10-66",
    "led_color": ["red"],
    "led_count": "0"
  }, {
    "eslId": "53-8F-F8-14",
    "led_color": ["blue"],
    "led_count": "0",
    "led_on_time": "10",
    "led_off_time": "30",
    "led_sleep_time": "40"
  }
]
```

3.7.4.1.3 切页请求示例

```
POST http://192.168.1.100:8080/prismart/setEslControl
[
  {
    "eslId": "59-12-10-66",
    "page_id": "1",
    "page_time": "3600"
  }
]
```

3.7.4.1.4 亮灯切页请求示例

```
POST http://192.168.1.100:8080/prismart/setEslControl
[
  {
    "eslId": "59-12-10-66",
    "page_id": "1",
    "page_time": "3600",
    "led_color": ["red"],
    "led_count": "20"
  }
]
```

3.7.4.2 返回示例

3.7.4.2.1 正常返回示例

```
{
  "resultCode": 1001,
  "result": "succeed",
  "message": "success",
  "data": [
    {
      "eslId": "59-12-10-66",
      "message": "success",
      "result": "succeed",
      "resultCode": 1001
    }
  ]
}
```

3.7.4.2.2 异常返回示例

```
{
  "resultCode": 1001,
  "result": "succeed",
  "message": "success",
  "data": [
    {
      "eslId": "59-12-10-66",
      "message": "最大不能超过255",
    }
  ]
}
```

```

        "result": "illegal_arguments",
        "resultCode": 1003
    }
]
}

```

3.8 getPr POST 查询绑定关系

3.8.1 API 说明

URL地址	http://{PriSmart访问地址}/getPr
METHOD	POST
内容类型	application/json;charset=utf-8
功能描述	通过价签ID或商品ID查询绑定关系。
需要身份认证	是。参考 2.2 身份认证 。
示例	http://192.168.1.100:8080/prismart/getPr

3.8.2 请求参数

List 类型，详细参数如下：

- **esId: String**。价签标识。与 **goodsId** 参数必填其一，两个参数都填，此参数优先极高。
- **goodsId: String**。商品 ID。与 **goodsId** 参数必填其一，两个参数都填， **esId** 参数优先极高。格式为：客户号/门店号/商品 SKU，例如 customer01/store01/1001。

3.8.3 返回参数

- **resultCode: Integer**。状态码。参考[2.3 返回参数说明](#)。
- **result: String**。返回结果。
- **message: String**。返回消息。
- **data: Map list**。返回数据内容。
 - **id: String**。价签标识
 - **esId: String**。价签标识

- **resultCode:** Integer。状态码。参考 [2.3 返回参数说明](#)。
- **extEsId:** String。价签 18 位扩展标识。
- **goodsSku:** String。商品 SKU。
- **goods:** Map。商品属性。
 - ◆ **customerStoreCode:** String。客户号。
 - ◆ **storeCode:** String。门店号。
 - ◆ **id:** String。商品 ID。
 - ◆ **sku:** String。商品 SKU。
 - ◆ **itemName:** String。商品名称。
 - ◆ **…:** 其他商品属性。
- **position:** Integer。分屏绑定时，绑定位置。
- **category:** Map。价签属性集合。
 - ◆ **firmware:** Integer。价签 FIRMWARE 标识。

3.8.4 示例

3.8.4.1 请求示例

3.8.4.1.1 以价签 ID 查询示例

```
[  
 {  
   "esId": "50-11-D5-EE"  
 }  
]
```

3.8.4.1.2 以商品 ID 查询示例

```
[  
 {  
   "goodsId": "customer01/store01/1001"  
 }  
]
```

3.8.4.2 返回示例

3.8.4.2.1 正常返回示例

```
{  
    "resultCode": 1001,  
    "result": "succeed",  
    "message": "success",  
    "data": [  
        {  
            "id": "50-11-D5-EE",  
            "resultCode": 1001,  
            "eslId": "50-11-D5-EE",  
            "extEslId": "",  
            "goodsSku": "11",  
            "goods": {  
                "customerStoreCode": "customer01",  
                "id": "customer01/store01/1001",  
                "storeCode": "store01",  
                "sku": "11",  
                ...  
            },  
            "position": 0,  
            "category": {  
                "firmware": 336708  
            }  
        }  
    ]  
}
```

3.8.4.2.2 异常返回示例

```
{  
    "resultCode" : 1003,  
    "result" : "illegal_arguments",  
    "message" : "Illegal arguments"  
}
```

3.9 integration POST 数据对接

3.9.1 API 说明

URL地址	http://{PriSmart访问地址}/integration
METHOD	POST
内容类型	application/json; charset=utf-8
功能描述	实现商品数据的更新、删除、价签绑定、解绑、闪灯和切页功能。
需要身份认证	否
示例	http://192.168.1.100:8080/prismart/integration

3.9.2 请求参数

此接口中，不同的对接指令请求的参数不同。

- **customerStoreCode:** String。必填。客户号。
- **storeCode:** String。必填。门店号。
- **batchNo:** String。必填。对接批次号。
- **callbackUrl:** String。选填。对接结果回调地址。默认最大长度 128 个字符，可通过修改数据库字段长度来调整（storedb 的 hs_integration_task 和 hs_integration_history 表的 back_url 字段）。系统默认未开启回调功能，使用时需在系统配置或门店配置中开启 CDI_CALLBACK 开关。
- **items:** Map list。对接指令数据。
 - **IIS_COMMAND:** String。必填。操作类型。取值说明如下：
 - ◆ **UPDATE:** 添加新商品，或者对已有商品数据进行增量更新，即只更新接口中提供的商品属性，未提供的属性保持原有值不变。当商品绑定了价签时，同时下发价签更新。
 - ◆ **COMPLETE_UPDATE:** 对商品数据进行全部属性的覆盖更新，对于接口中未提供的商品属性，将被置为默认值。当商品绑定了价签时，同时下发价签更新。
 - ◆ **BIND:** 根据商品 SKU 和价签标识绑定商品和价签。
 - ◆ **UNBIND_GOODS:** 根据商品 SKU 解除价签和商品的绑定关系。

- ◆ **UNBIND_ESLID:** 根据价签标识解除价签和商品的绑定关系。
- ◆ **UNBIND_DELETE_GOODS:** 根据商品 SKU 解除价签和商品的绑定关系，同时删除该商品。
- ◆ **CUTPAGE_FLASHLIGHTS:** 根据价签标识或商品 SKU 控制价签切页和/或亮灯。若同时提供价签标识和商品 SKU，系统会同时对商品 SKU 绑定的价签以及价签标识对应的价签执行指令。

当 **IIS_COMMAND** 取值 **UPDATE** 或 **COMPLETE_UPDATE** 时，参数如下：

- **customerStoreCode:** String。必填。客户号。
- **storeCode:** String。必填。门店号。
- **sku:** String。必填。商品 SKU。
- **ean:** String。选填。商品 EAN。
- **itemName:** String。选填。商品名称。
- …：其他商品参数。

当 **IIS_COMMAND** 取值 **BIND** 时，参数如下：

- **customerStoreCode:** String。必填。客户号。
- **storeCode:** String。必填。门店号。
- **esId:** String。必填。价签标识。
- **sku:** String。必填。商品 SKU。
- **ean:** String。选填。商品 EAN。

当 **IIS_COMMAND** 取值 **UNBIND_GOODS** 时，参数如下：

- **customerStoreCode:** String。必填。客户号。
- **storeCode:** String。必填。门店号。
- **sku:** String。必填。商品 SKU。
- **ean:** String。选填。商品 EAN。

当 **IIS_COMMAND** 取值 **UNBIND_ESLID** 时，参数如下：

- **customerStoreCode:** String。必填。客户号。
- **storeCode:** String。必填。门店号。
- **esId:** String。必填。价签标识。

当 **IIS_COMMAND** 取值 **UNBIND_DELETE_GOODS** 时，参数如下：

- **customerStoreCode:** String。必填。客户号。
- **storeCode:** String。必填。门店号。
- **sku:** String。必填。商品 SKU。

当 **IIS_COMMAND** 取值 **CUTPAGE_FLASHLIGHTS** 时，参数如下：

- **customerStoreCode:** String。必填。客户号。
- **storeCode:** String。必填。门店号。
- **sku:** String。必填。商品 SKU。
- **esId:** String。必填。价签标识。
- **IIS_PARAM:** Map。选填。闪灯切页参数。
 - ◆ **led_color:** String 数组。选填。闪灯颜色。传递多种颜色时会按照颜色依次展示。当前支持颜色值包括：blue、red、violet、green、indigo、yellow、white。
 - ◆ **led_count:** String。选填。闪灯循环次数。取值范围 0-65535。值为 0 时，表示灭灯。
 - ◆ **led_on_time:** String。选填。灯亮时长。取值范围 0-255，单位 30ms。
 - ◆ **led_off_time:** String。选填。灯灭时长。取值范围 0-255，单位 30ms。
 - ◆ **led_sleep_time:** String。选填。两次循环间隔时间。取值范围 0-65535，单位 30ms。
 - ◆ **page_id:** String。选填。切页时，目标页 ID。取值范围 0-7。
 - ◆ **page_time:** String。选填。在目标页的停留时间。取值范围 0-65535。单位 s。停留时间结束后会切换回模板中的默认页。取值为 0 时，表示不执行切页动作。

△ 注意：外层结点和子嵌套结点 **items** 结点必须传 **customerStoreCode** 与 **storeCode** 参数，否则会导致接口调用失败。

3.9.3 返回参数

- **resultCode:** Integer。状态码。参考 [2.3 返回参数说明](#)。
- **result:** String。返回结果。
- **message:** String。返回消息。

- **data:** Map list。返回数据内容。

- **esId:** String。价签标识。
- **resultCode:** Integer。状态码。参考 [2.3 返回参数说明](#)。
- **result:** String。返回结果。
- **message:** String。返回消息。

3.9.4 示例

3.9.4.1 请求示例

3.9.4.1.1 更新或添加商品示例

```
POST http://192.168.1.100:8080/prismart/integration
{
    "customerStoreCode": "customer01",
    "storeCode": "store01",
    "batchNo": "2-015c9bc5c9ca00001",
    "callbackUrl" : "",
    "items": [
        {
            "sku":"101",
            "ean":"1,2,3",
            "itemName": "红苹果",
            "IIS_COMMAND":"UPDATE",
            "price1":"101",
            "customerStoreCode": "customer01",
            "storeCode": "store01"
        }
    ]
}
```

3.9.4.1.2 全量更新或添加商品示例

```
POST http://192.168.1.100:8080/prismart/integration
{
    "customerStoreCode": "customer01",
    "storeCode": "store01",
    "batchNo": "2-015c9bc5c9ca00001",
    "callbackUrl" : "",
```

```
"items": [
  {
    "sku": "101",
    "ean": "1,2,3",
    "itemName": "红苹果",
    "IIS_COMMAND": "COMPLETE_UPDATE",
    "price1": "101",
    "customerStoreCode": "customer01",
    "storeCode": "store01"
  }
]
```

3.9.4.1.3 商品绑定示例

```
POST http://192.168.1.100:8080/prismart/integration
{
  "customerStoreCode": "customer01",
  "storeCode": "store01",
  "batchNo": "9877-2019042201",
  "items": [
    {
      "customerStoreCode": "customer01",
      "storeCode": "store01",
      "sku": "11",
      "ean": "6902265170501",
      "eslId": "55-2E-18-1C",
      "IIS_COMMAND": "BIND"
    }
  ]
}
```

3.9.4.1.4 商品解绑示例

```
POST http://192.168.1.100:8080/prismart/integration
{
  "customerStoreCode": "customer01",
  "storeCode": "store01",
  "batchNo": "2-015c9bc5c9ca00001",
  "items": [
    {

```

```
        "sku": "101",
        "ean": "1,2,3",
        "IIS_COMMAND": "UNBIND_GOODS",
        "customerStoreCode": "customer01",
        "storeCode": "store01"
    }
]
}
```

3.9.4.1.5 价签解绑示例

```
POST http://192.168.1.100:8080/prismart/integration
{
    "customerStoreCode" : "customer01",
    "storeCode" : "store01",
    "batchNo" : "9877-2019042201",
    "items" : [
        {
            "customerStoreCode" : "customer01",
            "storeCode" : "store01",
            "eslId": "55-2E-18-1C",
            "IIS_COMMAND": "UNBIND_ESLID"
        }
    ]
}
```

3.9.4.1.6 解绑并删除商品示例

```
POST http://192.168.1.100:8080/prismart/integration
{
    "customerStoreCode" : "customer01",
    "storeCode" : "store01",
    "batchNo" : "9877-2019042201",
    "items" : [
        {
            "customerStoreCode" : "customer01",
            "storeCode" : "store01",
            "sku" : "11",
            "IIS_COMMAND": "UNBIND_DELETE_GOODS"
        }
    ]
}
```

3.9.4.1.7 控制价签闪灯切页示例

```
POST http://192.168.1.100:8080/prismart/integration
{
    "customerStoreCode": "customer01",
    "storeCode": "store01",
    "batchNo": "3-015c9bc5c9ca0035",
    "items": [
        {
            "customerStoreCode": "customer01",
            "storeCode": "store01",
            "sku": "11",
            "IIS_COMMAND": "CUTPAGE_FLASHLIGHTS",
            "eslId": "55-62-89-1C",
            "IIS_PARAM": {
                "page_id": "2",
                "page_time": "30",
                "led_count": "10",
                "led_color": ["green"]
            }
        }
    ]
}
```

3.9.4.2 返回示例

3.9.4.2.1 正常返回示例

```
{
    "resultCode": 1001,
    "result": "succeed",
    "message": "success",
    "data": "INTEGRATION_CODE_SUCCESS"
}
```

3.9.4.2.2 异常返回示例

```
{
    "resultCode": 1102,
    "result": "illegal_account_status",
    "message": "Nonexistent store customer.1"
}
```

3.10 integration/flash POST 控制门店全部价签闪灯/灭灯

3.10.1 API 说明

URL地址	http://{PriSmart访问地址}/integration/flash
METHOD	POST
内容类型	application/json; charset=utf-8
功能描述	通过客户号和门店号对门店进行全局闪灯/灭灯。
需要身份认证	否
示例	http://192.168.1.100:8080/prismart/integration/flash

3.10.2 请求参数

Map 类型。详细参数如下：

- **batchId:** String。必填。请求批次号。
- **customerCode:** String。必填。客户号。
- **storeCode:** String。必填。门店号。
- **lightFlag:** Integer。必填。闪灯类型。0:灭灯； 1:亮灯

3.10.3 返回参数

- **resultCode:** Integer。状态码。参考 [2.3 返回参数说明](#)。
- **result:** String。返回结果。
- **message:** String。返回消息。
- **data:** String。返回数据内容。当请求异常时，此字段不返回。

3.10.4 示例

3.10.4.1 请求示例

```
{
    "batchId": "015c9bc5c9ca00001",
    "lightFlag": 0,
    "storeCode": "store01",
    "customerCode": "customer01"
}
```

3.10.4.2 返回示例

3.10.4.2.1 正常返回示例

```
{  
    "resultCode": 1001,  
    "result": "succeed",  
    "message": "success",  
    "data": "INTEGRATION_CODE_SUCCESS"  
}
```

3.10.4.2.2 异常返回示例

```
{  
    "resultCode": 1102,  
    "result": "illegal_account_status",  
    "message": "Nonexistent store customer01.store02"  
}
```