



# PcPos REST API

طبقه بندی: عادی

تهیه و تدوین: معاونت فناوری اطلاعات

این سند توسط شرکت پرداخت الکترونیک سداد تولید شده است و به عنوان دارایی با ارزش سازمان محسوب می شود.

هرگونه کپی برداری و یا استفاده دیگر از این سند بدون کسب مجوز کتبی از این شرکت ممنوع می باشد. ضمناً دارنده این سند از هرگونه افشاء یا انتشار سند به اشخاص دیگر جهت هرگونه استفاده و بهره

برداری بدون اطلاع این شرکت پرهیز نماید.

## تاریخچه مستند

شماره ویرایش	تاریخ صدور	شرح	نام و نام خانوادگی
۱.۰۰	۱۳۹۵/۱۲/۱۵	تدوین	علی گیتی نژاد
۱.۰۱	۱۳۹۶/۰۴/۲۱	ویرایش	احمد بابایی کفشگر
۱.۰۲	۱۳۹۶/۰۸/۱۵	ویرایش	احمد بابایی کفشگر
۱.۰۳	۱۳۹۶/۰۸/۲۹	ویرایش	احمد بابایی کفشگر
۱.۰۴	۱۳۹۶/۰۹/۰۹	ویرایش	احمد بابایی کفشگر
۱.۰۵	۱۳۹۶/۱۲/۱۵	ویرایش	احمد بابایی کفشگر
۱.۰۶	۱۳۹۷/۰۲/۲۲	ویرایش	احمد بابایی کفشگر
۱.۰۷	۱۳۹۷/۰۶/۱۲	اضافه شدن OrderId برای هماهنگی و پیگیری هر تراکنش اضافه شدن قابلیت لغو تراکنش قابلیت تنظیم RetryTimeout و ResponseTimeout در ورودی	احمد بابایی کفشگر
۱.۰۸	۱۳۹۷/۰۶/۲۶	اضافه شدن گرفتن خصوصیات Config در سرویس REST اضافه شدن گرفتن مشخصات پوز	احمد بابایی کفشگر
۱.۰۹	۱۳۹۷/۰۷/۰۳	حذف RequestId اصلاح نمونه کدها اضافه شدن تبدیل کدهای برگشتی از سوئیچ به پیام فارسی	احمد بابایی کفشگر
۱.۱۰	۱۳۹۷/۰۸/۲۷	اضافه شدن فیلدهای جدید به GetCardInfo	احمد بابایی کفشگر
۱.۱۱	۱۳۹۷/۱۰/۰۱	تغییرات خصوصیات کلاس‌ها اضافه شدن تراکنش‌های بسته حمایت و امنیت غذایی	احمد بابایی کفشگر
۱.۱۲	۱۳۹۷/۱۰/۱۰	بروزرسانی اطلاعات تسهیم	احمد بابایی کفشگر
۱.۱۳	۱۳۹۷/۱۱/۲۸	اصلاح نکات و راهنمایی	احمد بابایی کفشگر
۱.۱۴	۱۳۹۷/۱۲/۱۴	تغییر جستجوی لیست دستگاه‌ها، اضافه شدن شروع بکار مجدد	احمد بابایی کفشگر
۱.۱۵	۱۳۹۸/۰۴/۱۱	گزارش لیست حساب‌های متصل به دستگاه پوز جستجوی تراکنش با Order ID	احمد بابایی کفشگر
۱.۱۶	۱۳۹۸/۱۱/۱۶	اصلاح کنترل خطاها، تغییر مدلهای، بهبود ارتباط پوز مجیک	احمد بابایی کفشگر
۱.۱۷	۱۳۹۸/۱۲/۰۶	اصلاح کنترل خطاها، بهبود ارتباط پوز مجیک، استعلام تراکنش مجیک	محمد مهدی میردامادی
۱.۱۸	۱۳۹۸/۰۲/۱۶	اضافه شدن متن انتهای رسید	محمد مهدی میردامادی
۱.۱۹	۱۴۰۱/۰۸/۰۳	اضافه شدن کد اختصاصی نرم افزار	احمد بابایی کفشگر
۱.۲۰	۱۴۰۳/۱۱/۲۸	تغییرات مرتبط با MultiSaleId	احمد بابایی کفشگر

## فهرست مطالب

۵.....	۱ هدف
۵.....	۲ دامنه کاربرد
۵.....	۳ مسؤولیت ها
۶.....	۴ درباره PCPOS REST API
۶.....	۴.۱ محیط برنامه نویسی مورد نیاز برای توسعه ی نرم افزار با استفاده از REST API
۷.....	۵ متدها (Methods)
۷.....	۵.۱ GetInfo
۸.....	۵.۲ GetDevices
۹.....	۵.۳ Get Serial Ports
۱۰.....	۵.۴ Get Settings
۱۱.....	۵.۵ Sale Transaction
۱۵.....	۵.۶ Abort Transaction
۱۶.....	۵.۷ Device Info
۱۷.....	۵.۸ Bill Payment Transaction
۱۸.....	۵.۹ Identified Inquiry Transaction
۱۹.....	۵.۱۰ Government Identified Inquiry Transaction
۲۱.....	۵.۱۱ Commodity Basket Transaction
۲۲.....	۵.۱۲ Food Safety Transaction
۲۳.....	۵.۱۳ Transaction Reports
۲۵.....	۵.۱۴ Search
۲۶.....	۵.۱۵ Get Card Info
۲۷.....	۵.۱۶ Get Accounts
۲۸.....	۵.۱۷ شروع از نو سرویس REST
۲۸.....	۵.۱۸ Magic Inquiry
۳۰.....	۵.۱۹ جواب تراکنش
۳۳.....	۶ توضیحات، نکات و تنظیمات
۳۳.....	۶.۱ تست نرم افزار با REST Client

۶.۲	تنظیمات درخواست در ajax	۳۳
۶.۳	تنظیمات مربوط به پوزهای با قابلیت ارتباط سریال	۳۴
۶.۴	اجازه استفاده گواهینامه امنیتی در ویندوز	۳۵
۶.۵	دیواره آتش	۳۷

## ۱ هدف

ارایه مستندات REST API ارایه شده در PCPOS می باشد.

## ۲ دامنه کاربرد

این مستند به منظور آشنایی برنامه نویسان جهت استفاده از REST API پی سی پوز دستگاه های برند بلوبرد، بایتل و مجیک شرکت پرداخت الکترونیک سداد است.

برند	مدل
بلوبرد <sup>۱</sup>	P3500, CT280, MT280
بایتل <sup>۲</sup>	IC3300, IC3100
مجیک <sup>۳</sup>	X8, X5
کستلز <sup>۴</sup>	V5S
پکس <sup>۵</sup>	S800, Q80

## ۳ مسئولیت ها

در راه اندازی سرویس پی سی پوز همکاری واحد های پشتیبانی و بخش فناوری اطلاعات جهت راه اندازی الزامی میباشد

<sup>1</sup> BlueBird

<sup>2</sup> Bitel

<sup>3</sup> Magic

<sup>4</sup> Castles

<sup>5</sup> Pax

## ۴ درباره PCPOS REST API

نسخه 5.3.531.1632 رابط برنامه نویسی کاربردی PcPos REST به منظور توسعه ی نرم افزار و اتصال به دستگاه پوز در محیط های مبتنی بر REST طراحی شده است. ارتباط بین دستگاه پوز و سرویس PCPOS از طریق شبکه یا پورت سریال میسر می باشد. در ارتباط از طریق شبکه ابتدا شبکه جهت شناسایی پوزهای در دسترس جستجو می شود. سپس دستگاه مورد نظر جهت ادامه فرآیند خرید مورد استفاده قرار می گیرد. در ارتباط سریال هم کفایت بعد از مقدار دهی شماره (نام) پورت سریال ادامه فرآیند تراکنش انجام شود.

PCPOS REST API امکانات زیر را در اختیار کاربران قرار می دهد:

- امکان ارتباط با پروتکل HTTP و HTTPS (برای صحت عملکرد پروتکل HTTPS بخش نکات را بررسی نمایید)
- امکان شناسایی پوز های موجود در شبکه
- امکان ارسال تراکنش از طریق بستر سریال
- امکان ارسال تراکنش و دریافت نتیجه
- امکان لغو تراکنش از سمت رابط برنامه نویسی
- امکان لغو تراکنش از سمت پوز
- ارایه امکان خرید تک حسابی و چند حسابی
- امکان چاپ متن انتهای رسید دستگاه POS
- ارایه امکان پرداخت قبض
- امکان دریافت ۱۳ تراکنش آخر ثبت شده در دستگاه POS
- امکان جستجو با شناسه تراکنش در تراکنش های ثبت شده در دستگاه POS
- انتقال رمز گذاری شده اطلاعات بین پوز و DLL پی سی پوز
- گرفتن شماره کارت به صورت رمزنگاری شده
- ذخیره log خطاهای اتفاق افتاده در مسیر نصب
- استعمال RowIndex ثبت شده در مرکز به ازاء شماره شب (نیاز به اینترنت دارد).
- جستجوی تراکنشهای ذخیره شده در دستگاه بر اساس Order ID

### ۴/۱ محیط برنامه نویسی مورد نیاز برای توسعه ی نرم افزار با استفاده از REST API

با توجه به در دسترس بودن رابط برنامه نویسی کاربردی REST از طریق آدرس IP آن، از هر محیط برنامه نویسی ای که بتوان به این سرویس دسترسی داشت میتوان اقدام به توسعه نرم افزار کاربردی شخصی نمود. پیش نیاز این سرویس نسخه ۳.۵ از ابزار .Net Framework، سیستم عامل Windows XP SP3 به بعد می باشد.

## ۵ متدها (Methods)

### GetInfo ۵/۱

Prototype:	<a href="http://localhost:8000/api/GetInfo">http://localhost:8000/api/GetInfo</a>
Function:	مشخصات رابط برنامه نویسی برگردانده می شود.
Be Applicable In:	All
Notes:	این متد فقط برای تست دسترسی یا اجرای دستور توسط سرویس REST میباشد.
Request:	<a href="http://localhost:8000/api/GetInfo">http://localhost:8000/api/GetInfo</a>
HTTP Method:	Get
Body:	null
Response:	<pre>{   "Description": "This is REST API service for SadadPSP PCPOS.",   "Supported Devices": "BlueBird P3500, BlueBird CT280, BlueBird MT280, Bitel IC3300, Bitel IC3100, Magic X8, Magic X5",   "Version": "a.b.c.d" }</pre>

## GetDevices ۵/۲

Prototype:	<code>http://localhost:8000/api/GetDevices/{port}/{timeout}</code>
Function:	لیست دستگاه های POS موجود در شبکه LAN و متصل شده به پورت های سریال جهت ارسال و دریافت داده به خروجی برگردانده می شود.
Be Applicable In:	BlueBird {P3500, CT280, MT280}
Notes:	پیش فرض پارامتر {port} مقدار 100 میباشد. مقدار پارامتر {port} به پورت تنظیم شده در دستگاه POS وابستگی مستقیم دارد و مقدار آن 100 است. پیش فرض پارامتر {timeout} مقدار 1000 میباشد. مقدار پارامتر {timeout} باید عددی بین 1000 و 300000 (معادل ۵ دقیقه) باشد. اگر مقدار پارامترها را 0 وارد کنیم مقادیر پیش فرض به آن اختصاص داده می شود. <b>زمان پاسخ دهی این متد بستگی به Timeout تنظیم شده دارد و ممکن است بیش از ۱۰ ثانیه طول بکشد.</b> در خصوصیات پاسخ این متد میتوان موارد مورد نیاز برای اتصال به دستگاه POS را گرفت.
Request:	<code>http://localhost:8000/api/GetDevices/0/0</code>
HTTP Method:	Get
Body:	null
Response:	<pre>[   {     "ComPort": "",     "ConnectionType": 1,     "ConnectionTypeName": "Lan",     "DeviceType": 0,     "DeviceTypeName": "BlueBird",     "Header": "SADAD PCPOS",     "IpAddress": "172.24.32.186",     "MerchantId": "000000000000152",     "MerchantName": "پوز ۱۸۶",     "Port": "8888",     "TerminalId": "C0000301"   },   {     "ComPort": "COM8",     "ConnectionType": 0,     "ConnectionTypeName": "Serial",     "DeviceType": 0,     "DeviceTypeName": "BlueBird",     "Header": "SADAD PCPOS",     "IpAddress": "0.0.0.0",     "MerchantId": "000000130325369",     "MerchantName": "تست پشتیبانی",     "Port": "0000",     "TerminalId": "23406522"   } ]</pre>



## Get Serial Ports ۵/۳

Prototype:	<a href="http://localhost:8000/api/GetSerialPorts">http://localhost:8000/api/GetSerialPorts</a>
Function:	لیست پورتهای فعال به رایانه ای که سرویس در آن اجرا شده است را جهت ارسال و دریافت داده به خروجی برگردانده می شود.
Be Applicable In:	All
Notes:	دستگاه POS نیز باید از طریق پورت سریال به رایانه مورد نظر متصل باشد.
Request:	<a href="http://localhost:8000/api/GetSerialPorts">http://localhost:8000/api/GetSerialPorts</a>
HTTP Method:	Get
Body:	null
Response:	[ "COM1", "COM5", "COM6" ]

## Get Settings ۵/۴

Prototype:	<a href="http://localhost:8000/api/GetSettings">http://localhost:8000/api/GetSettings</a>
Function:	لیست تنظیمات در فایل Config سرویس را در قالب فایل json به صورت Key, Value برگردانده می‌شود. لطفاً قسمت ۶.۳ مطالعه شود.
Be Applicable In:	All
Notes:	این فایل توسط پشتیبان شرکت سداد باید تنظیم گردد. در صورت نیاز توسعه دهندگان آشنا نیز قادر به تغییر آن در سرویس می‌باشند. تغییرات اشتباه در این فایل در کار سرویس تداخل ایجاد میکند بنابراین مسئولیت اتفاقات احتمالی با فرد تغییر دهنده می‌باشد. در دستگاه‌هایی که با درگاه سریال یا COM به رایانه متصل هستند نیاز است ابتدا شماره یا نام درگاه را در رایانه چک کنند و معادل آن را در فایل Config وارد کنند. مقادیر مورد پذیرش برای DeviceType: 0=BlueBird, 1=Magic, 2=Bitel, 3=Pax, 4=Castle می‌باشد.
Request:	<a href="http://localhost:8000/api/GetSettings">http://localhost:8000/api/GetSettings</a>
HTTP Method:	Get
Body:	null
Response:	[       {         "LockItem": false,         "DeviceType": "0",         "DeviceTypeName": "BlueBird",         "MerchantId": "01010101010",         "SerialPort": "COM8",         "Tag": "pos1",         "TerminalId": "001",         "BaudRate": "115200",         "StopBits": "One"       },       {         "LockItem": false,         "DeviceType": "0",         "DeviceTypeName": "BlueBird",         "MerchantId": "65463102361",         "SerialPort": "COM9",         "Tag": "pos2",         "TerminalId": "001",         "BaudRate": "115200",         "StopBits": "One"       }     ]

## ۵/۵ Sale Transaction

نحوه انجام تراکنش از این متد به بعد در این تراکنش بدین صورت است که ورودی ها در بدنه (Body) درخواست فرستاده می شود. توجه داشته باشید روال کار بدین صورت است که ابتدا خصوصیات مورد نیاز از متدهای GetSerialPorts یا GetDevices گرفته و مقادیر آنها را در بدنه توابعی که نیاز دارند جایگذاری می کنیم. برای تمامی متدهای درخواست از POS به موارد زیر نیاز است:

اگر اتصال از PC به POS از نوع سریال باشد:

- شماره پورت سریال (SerialPort)
- ConnectionType = "Serial"
- مثال JSON:

```
{"SerialPort":"COM3","ConnectionType":"Serial","MultiAccount":"1:10,2:30,4:40,3:20","DivideType":"3","Amount":"50000"}
```

و اگر اتصال از PC به POS از نوع LAN باشد:

- آدرس IP (DeviceIp)
- شماره پورت (DevicePort)
- ConnectionType = "LAN"
- مثال JSON:

```
{"DeviceIp":"172.24.32.186","DevicePort":"8888","ConnectionType":"Lan","MultiAccount":"1:10,2:30,4:40,3:20","DivideType":"3","Amount":"50000"}
```

موارد دیگری به عنوان ورودی اختیاری در نظر گرفته شده اند. این موارد بسته به نیاز مشتری می توانند همراه هر درخواست POST مقداردهی و ارسال شوند:

- تسهیم: تسهیم در پی سی پوز به دو صورت درصدی و مبلغی قابل انجام است. برای انجام این مورد نیاز است مشتری (صاحب دستگاه پوز) درخواست کتبی افزایش حساب به همراه شماره حساب ها را به ازاء ترمینال (Terminal) یا مشتری (Merchant) به شرکت پرداخت الکترونیک سداد تسلیم کند تا عملیات مورد نظر انجام پذیرد. بعد از انجام این مورد، میتوان از مرکز، ترمینال (پوز) را در حالت تک حسابی یا چند حسابی تنظیم کرد.

حال فرض می کنیم ترمینال در حالت چند حسابی باشد و در تراکنش باید تسهیم انجام پذیرد. برای این کار باید شماره اندیس هر شماره حساب ثبت شده برای مشتری در سیستم را بدانیم که به دو صورت قابل انجام است:

۱. بعد از ارائه درخواست کتبی افزایش حساب شماره اندیس ها در اختیار مشتری قرار می گیرد.

۲. از طریق وب سرویس (نیاز به اتصال اینترنت دارد) و ورود شماره شبا قابل دسترسی است.

بنابراین برای ورود داده تسهیم به دو روش ورود شماره اندیس و یا شماره شبا بجای اندیس میتوان عمل کرد که در مثال آمده است.

توسعه دهندگان و مشتریانی که از این سرویس استفاده می کنند توجه داشته باشند اگر در جواب تراکنش خطای شماره ۳۰ (فرمت اشتباه است) دریافت می کنند به یکی از موارد زیر مرتبط می باشد:

۱. اگر ترمینال (پوز) تک حسابی باشد و از پی سی پوز اطلاعات تسهیم فرستاده شود.

۲. اگر ترمینال (پوز) چند حسابی باشد و از پی سی پوز اطلاعات تسهیم پر نشود.

۳. اگر ترمینال (پوز) تک حسابی باشد و از پی سی پوز اطلاعات تسهیم عمدا یا سهوا اشتباه پر شود. مثلا شماره اندیس حساب اشتباه وارد

شود، یا اندیسی وارد شود که وجود ندارد، یا شماره شبای حسابی وارد شود که در سوئیچ پرداخت ثبت نشده است، یا مجموع درصد برابر با

- ۱۰۰ نباشد، و یا مجموع مبالغ تسهیم با مبلغ تراکنش برابر نباشد.
  - موارد متعددی پیش آمده است که مشتری درخواست افزایش حساب و تبدیل دستگاه به چندحسابی به شرکت پرداخت الکترونیک سداد نداده است ولی در تراکنش اطلاعات تسهیم را میفرستد یا بالعکس.
  - شناسه تراکنش (OrderId): به عنوان کلید واحدی برای تراکنش‌های درخواست داده شده به دستگاه پوز در نظر گرفته شده است.
  - مشتری در جواب تراکنش مقدار این خصوصیت را دریافت می‌کند تا اعتبارسنجی ای برای تراکنش‌های داشته باشد.
  - مدت زمان انتظار برای تکرار ارسال درخواست (RetryTimeout): این خصوصیت برای مدت زمان انتظار برای تکرار ارسال درخواست در صورت نیاز است.
  - مدت زمان انتظار برای دریافت پاسخ تراکنش (ResponseTimeout): این خصوصیت برای مدت زمان انتظار برای دریافت جواب تراکنش از سمت دستگاه پوز است.
- مثال:

```
{"OrderId":"435261","RetryTimeOut":"5000, 5000, 5000","ResponseTimeout":"180000, 5000, 5000"}
```

- نوع دستگاه (DeviceType): این خصوصیت برای ارتباط با دستگاه تعبیه شده است و به ازاء دستگاه‌های پوز رفتار و ارتباط متفاوتی ایجاد میشود که از دید نرم افزار های شخص سوم تفاوتی محسوس نمی‌باشد. اگر این خصوصیت در ورودی وجود نداشته باشد، اطلاعات مورد نیاز از فایل Config خوانده شده و نوع ارتباط تعیین میگردد، لذا نیاز است این تنظیمات در فایل Config به درستی انجام پذیرد.
- برای هر یک از دستگاه ها کدی در نظر گرفته شده است که به این قرار می‌باشد:

- کد ۰: دستگاه با برند BlueBird
- کد ۱: دستگاه با برند Magic
- کد ۲: دستگاه با برند Bitel
- کد ۳: دستگاه با برند Pax
- کد ۴: دستگاه با برند Castle

Prototype:	<a href="http://localhost:8000/api/Sale">http://localhost:8000/api/Sale</a>
Function:	عملیات خرید
Be Applicable In:	A11
Notes:	<ul style="list-style-type: none"> <li>مقدار Content-Type = application/json شود.</li> <li>مقدار MultiAccount طبق استاندارد ی که توضیح داده شده پر گردد: {ARN1,Percent1/Amount1;ARN2,Percent2/Amount2;...} '/' به معنی OR میباشد، یعنی یا درصد یا مبلغ را وارد میکنیم.</li> <li>مقدار DivideType مقدارهای ۳ (تقسیم درصدی) و ۴ (تقسیم مبلغی) می تواند باشد.</li> <li>مثال ۱: اگر مقدار DivideType = 3 باشد مقدار MultiAccount می‌شود: 1:10,2:30,4:40,6:20</li> <li>مثال ۲: اگر مقدار DivideType = 4 باشد و مبلغ ۲۰۰۰۰۰، مقدار MultiAccount می‌شود: 1:100,2:3000,4:400,6:196500</li> </ul>

	<ul style="list-style-type: none"> <li>مقدار AdvertisementData جهت چاپ داده های تبلیغاتی و متفرقه می تواند استفاده شود. برای رفتن به خط بعد از کاراکتر "،" استفاده گردد.</li> </ul>
Request:	http://localhost:8000/api/Sale
HTTP Method:	POST
Body:	<pre>{   "DeviceType": "0",   "DeviceIp": "172.24.32.212",   "DevicePort": "8888",   "ConnectionType": "Lan",   "OrderId": "435261",   "RetryTimeout": "5000, 5000, 5000",   "ResponseTimeout": "180000, 5000, 5000",   "TerminalId": "C0000301",   "MerchantId": "M00000000000152",   "MultiAccount": "IR350170000000106928854003:10, IR350170000000106928854006:30, IR4301700000002171291343008:40, IR4301700000002171291343005:20",   "DivideType": "3",   "Amount": "50000",   "AdvertisementData": "نام: علی, نام خانوادگی: محمدی",   "AppCode": 12345 }</pre> <p>Or</p> <pre>{   "DeviceType": "0",   "DeviceIp": "172.24.32.212",   "DevicePort": "8888",   "OrderId": "435261",   "RetryTimeout": "5000, 5000, 5000",   "ResponseTimeout": "180000, 5000, 5000",   "TerminalId": "C0000301",   "MerchantId": "M00000000000152",   "ConnectionType": "Lan",   "MultiAccount": "1:10,2:30,4:40,6:20",   "DivideType": "3",   "Amount": "50000",   "AdvertisementData": "نام: علی, نام خانوادگی: محمدی",   "AppCode": 12345 }</pre>
Response:	<pre>{   "Amount": "50000",   "ApprovalCode": "0014 ",   "CardNo": "621986*****386",   "Merchant": "M00000000000152",   "OptionalField": "",   "OrderId": "435261",   "PacketType": null,   "PcPosStatus": "ارتباط موفق با دستگاه",   "ProcessingCode": "000000",   "ResponseCode": "00",   "Rrn": "320047611089",   "Terminal": "C0000301",   "TransactionDate": "1396/04/03",   "TransactionNo": "001139",   "TransactionTime": "14:07:29" }</pre>
Body: (Magic	{
مهر کیفیت:	این سند توسط شرکت پرداخت الکترونیک سداد تولید شده است. هرگونه کپی برداری و یا استفاده دیگر از این سند، بدون کسب مجوز کپی از این شرکت ممنوع می باشد.
صفحه 13 از ۳۷	

sample)	<pre> "DeviceType" : "1", "TerminalId" : "001", "MerchantId" : "011900133", "Amount" : "80000", "ConnectionType" : "Serial", "SerialPort" : "COM1", "SaleId" : "123456" } </pre>
Response: (Magic sample)	<pre> { "PcPosStatus": "ارتباط موفق با دستگاه", "PcPosStatusCode": 4, "ResponseCode": "00", "ResponseCodeMessage": "تراکنش موفق", "Amount": "80000", "ApprovalCode": "", "CardNo": "589210*****1", "MerchantId": "011900133", "OptionalField": "123456", "OrderId": "", "PacketType": "", "ProcessingCode": "", "Rrn": "144700170050", "SaleId": "", "TerminalId": "37914470", "TransactionDate": "1398/11/30", "TransactionNo": "501700", "TransactionTime": "11:24:24" } </pre>

## Abort Transaction ۵/۶

Prototype:	<a href="http://localhost:8000/api/Abort">http://localhost:8000/api/Abort</a>
Function:	لغو عملیات.
Be Applicable In:	All
Notes:	در صورت نمایش پیغام "خطای درخواست همزمان" میتوان از این متد برای لغو عملیات قبلی استفاده کرد.
Request:	<a href="http://localhost:8000/api/Abort">http://localhost:8000/api/Abort</a>
HTTP Method:	POST
Body:	<pre>{   "ConnectionType" : "Lan",   "DeviceType": "0",   "DeviceIp": "172.24.32.212",   "DevicePort": "8888" } Or {   "ConnectionType" : "Serial",   "DeviceType": "0",   "SerialPort": "COM1" }</pre>
Response:	<pre>{   "ResponseCode": "0",   "ResponseCodeMessage": "تراکنش موفق" }</pre>

## Device Info ۵/۷

Prototype:	<a href="http://localhost:8000/api/Deviceinfo">http://localhost:8000/api/Deviceinfo</a>
Function:	این متد خصوصیات دستگاه را به خروجی برمی گرداند.
Be Applicable In:	All
Notes:	<p>جواب این متد در هر یک از دستگاه های پوز متفاوت میباشد و بدین صورت است:</p> <p>۱. در پوزهای BlueBird جواب از سمت پوز می باشد.</p> <p>۲. در پوزهای Magic خصوصیات مورد نیاز از فایل Config خوانده و برگردانده می شود.</p>
Request:	<a href="http://localhost:8000/api/DeviceInfo">http://localhost:8000/api/DeviceInfo</a>
HTTP Method:	POST
Body:	<pre>{   "SerialPort": "COM1",   "ConnectionType": "Serial",   "DeviceType": "1",   "OrderId": "9512",   "TerminalId": "C0000301",   "MerchantId": "000000000000301" }</pre>
Response:	<pre>{   "DeviceType": "1",   "DeviceTypeName": "Magic",   "MerchantId": "00000301",   "TerminalId": "001" } Or {   "DeviceType": "0",   "DeviceTypeName": "BlueBird",   "Header": "SADAD PCPOS",   "IpAddress": "172.24.32.178",   "MerchantId": "000000000000301",   "MerchantName": "پرداخت الکترونیک سداد",   "Port": "8888",   "TerminalId": "2334578" }</pre>



## Bill Payment Transaction ۵/۸

Prototype:	<a href="http://localhost:8000/api/BillPayment">http://localhost:8000/api/BillPayment</a>
Function:	عملیات پرداخت قبض.
Be Applicable In:	BlueBird {P3500, CT280, MT280}, Bitel {IC3300, IC3100}
Notes:	<ul style="list-style-type: none"> <li>مقدار AdvertisementData جهت چاپ داده های تبلیغاتی و متفرقه می تواند استفاده شود. برای رفتن به خط بعد از کاراکتر "،" استفاده گردد.</li> </ul>
Request:	<a href="http://localhost:8000/api/BillPayment">http://localhost:8000/api/BillPayment</a>
HTTP Method:	POST
Body:	<pre>{   "DeviceIp": "172.24.32.212",   "DevicePort": "8888",   "ConnectionType": "Lan",   "DeviceType": "0",   "OrderId": "435261",   "RetryTimeout": "5000, 5000, 5000",   "ResponseTimeout": "180000, 5000, 5000",   "BillId": "6039628301226",   "PayId": "189840835",   "AdvertisementData": "نام: علی, نام خانوادگی: محمدی",   "AppCode": 12345 }</pre>
Response:	<pre>{   "Amount": "1898000",   "ApprovalCode": "0014",   "CardNo": "621986*****386",   "Merchant": "M00000000000152",   "OptionalField": "",   "OrderId": "435261",   "PacketType": null,   "PcPosStatus": "ارتباط موفق با دستگاه",   "ProcessingCode": "170000",   "ResponseCode": "00",   "Rrn": "320047611093",   "Terminal": "C0000301",   "TransactionDate": "1396/04/03",   "TransactionNo": "001140",   "TransactionTime": "15:28:08" }</pre>

## Identified Inquiry Transaction ۵/۹

Prototype:	<a href="http://localhost:8000/api/IdentifiedInquiry">http://localhost:8000/api/IdentifiedInquiry</a>
Function:	عملیات خرید شناسه دار با استعلام.
Be Applicable In:	BlueBird {P3500, CT280, MT280}, Bitel {IC3300, IC3100}
Notes:	<ul style="list-style-type: none"> <li>مقدار AdvertisementData جهت چاپ داده های تبلیغاتی و متفرقه می تواند استفاده شود. برای رفتن به خط بعد از کاراکتر "," استفاده گردد.</li> </ul>
Request:	<a href="http://localhost:8000/api/IdentifiedInquiry">http://localhost:8000/api/IdentifiedInquiry</a>
HTTP Method:	POST
Body:	<pre>{   "DeviceIp": "172.24.32.212",   "DevicePort": "8888",   "ConnectionType": "Lan",   "DeviceType": "0",   "OrderId": "435261",   "RetryTimeout": "5000, 5000, 5000",   "ResponseTimeout": "180000, 5000, 5000",   "InquiryId": "139522000000014335",   "AdvertisementData": "مشخصات پرداخت کننده, نام: علی, نام خانوادگی: محمدی",   "AppCode": 12345 }</pre>
Response:	<pre>{   "Amount": "1898000",   "ApprovalCode": "0014",   "CardNo": "621986*****386",   "Merchant": "M00000000000152",   "OptionalField": "",   "OrderId": "435261",   "PacketType": null,   "PcPosStatus": "ارتباط موفق با دستگاه",   "ProcessingCode": "170000",   "ResponseCode": "00",   "Rrn": "320047611093",   "Terminal": "C0000301",   "TransactionDate": "1396/04/03",   "TransactionNo": "001140",   "TransactionTime": "15:28:08" }</pre>

## Government Identified Inquiry Transaction ۵/۱۰

Prototype:	<a href="http://localhost:8000/api/GovernmentIdentifiedInquiry">http://localhost:8000/api/GovernmentIdentifiedInquiry</a>
Function:	عملیات خرید شناسه دار سازمانی با استعلام.
Be Applicable In:	BlueBird {P3500, CT280, MT280}, Bitel {IC3300, IC3100}
Notes:	<ul style="list-style-type: none"> <li>مقدار AdvertisementData جهت چاپ داده های تبلیغاتی و متفرقه می تواند استفاده شود. برای رفتن به خط بعد از کاراکتر "," استفاده گردد.</li> </ul>
Request:	<a href="http://localhost:8000/api/GovernmentIdentifiedInquiry">http://localhost:8000/api/GovernmentIdentifiedInquiry</a>
HTTP Method:	POST
Body:	<pre>{   "DeviceIp": "172.24.32.212",   "DevicePort": "8888",   "ConnectionType": "Lan",   "DeviceType": "0",   "OrderId": "435261",   "RetryTimeout": "5000, 5000, 5000",   "ResponseTimeout": "180000, 5000, 5000",   "InquiryId": "102110000220000121100000000000",   "Amount": "110000",   "AdvertisementData": "مشخصات پرداخت کننده, نام: علی, نام خانوادگی: محمدی",   "AppCode": 12345 }</pre> <p>برای خرید چندشناسه ای باید از الگوی زیر پیروی کرد:</p> <pre>{   "DeviceIp": "172.24.32.212",   "DevicePort": "8888",   "ConnectionType": "Lan",   "DeviceType": "0",   "OrderId": "435261",   "RetryTimeout": "5000, 5000, 5000",   "ResponseTimeout": "180000, 5000, 5000",   "InquiryIds": [     {       "Amount": 35000,       "Index": 1, // Or IBAN can be use as showing next properties       "IBAN": "IR123456789012345678901234",       "InquiryId": "102110000220000121100000000000"     }, {       "Amount": 140000,       "Index": 2, // Or IBAN can be use as showing next properties       "IBAN": "IR123456789012345678901235",       "InquiryId": "102110000220000121100000000001"     }   ],   "Amount": "175000",   "AdvertisementData": "مشخصات پرداخت کننده, نام: علی, نام خانوادگی: محمدی",   "AppCode": 12345 }</pre> <p>برای خرید چندشناسه ای و فایل SadadConfig.xml باید از الگوی زیر پیروی کرد:</p> <pre>{   "DeviceIp": "172.24.32.212",   "DevicePort": "8888",   "ConnectionType": "Lan",   "DeviceType": "0",</pre>

	<pre> "OrderId" : "435261", "RetryTimeout" : "5000, 5000, 5000", "ResponseTimeout" : "180000, 5000, 5000", "InquiryIds": [{   "Amount": 35000,   "InquiryId": "102110000220000121100000000000" }, {   "Amount": 140000,   "InquiryId": "102110000220000121100000000001" } ], "Amount" : "175000", "AdvertisementData" : "نام: علی, نام خانوادگی: محمدی", "AppCode" : 12345 } </pre>
Response:	<pre> {   "Amount": "000000110000",   "ApprovalCode": "",   "CardNo": "621986*****386",   "Merchant": "M00000000000152",   "OptionalField": "",   "OrderId" : "435261",   "PacketType": null,   "PcPosStatus": "ارتباط موفق با دستگاه",   "ProcessingCode": "270000",   "ResponseCode": "00",   "Rrn": "",   "Terminal": "C0000301",   "TransactionDate": "1396/04/14",   "TransactionNo": "000018",   "TransactionTime": "15:03:21" } </pre>

## Commodity Basket Transaction ۵/۱۱

Prototype:	<a href="http://localhost:8000/api/CommodityBasket">http://localhost:8000/api/CommodityBasket</a>
Function:	عملیات خرید بسته سید کالا.
Be Applicable In:	BlueBird {P3500, CT280, MT280}, Bitel {IC3300, IC3100}
Notes:	
Request:	<a href="http://localhost:8000/api/CommodityBasket">http://localhost:8000/api/CommodityBasket</a>
HTTP Method:	POST
Body:	<pre>{   "DeviceIp": "172.24.32.212",   "DevicePort": "8888",   "ConnectionType": "Lan",   "DeviceType": "0",   "OrderId": "435261",   "RetryTimeout": "5000, 5000, 5000",   "ResponseTimeout": "180000, 5000, 5000",   "Amount": "110000" }</pre>
Response:	<pre>{   "Amount": "000000110000",   "ApprovalCode": "",   "CardNo": "621986*****386",   "Merchant": "M00000000000152",   "OptionalField": "",   "OrderId": "435261",   "PacketType": null,   "PcPosStatus": "ارتباط موفق با دستگاه",   "ProcessingCode": "270000",   "ResponseCode": "00",   "Rrn": "",   "Terminal": "C0000301",   "TransactionDate": "1396/04/14",   "TransactionNo": "000018",   "TransactionTime": "15:03:21" }</pre>

## Food Safety Transaction ۵/۱۲

Prototype:	<a href="http://localhost:8000/api/FoodSafety">http://localhost:8000/api/FoodSafety</a>
Function:	عملیات خرید بسته امنیت غذایی.
Be Applicable In:	BlueBird {P3500, CT280, MT280}, Bitel {IC3300, IC3100}
Notes:	
Request:	<a href="http://localhost:8000/api/FoodSafety">http://localhost:8000/api/FoodSafety</a>
HTTP Method:	POST
Body:	<pre>{   "DeviceIp": "172.24.32.212",   "DevicePort": "8888",   "ConnectionType": "Lan",   "DeviceType": "0",   "OrderId": "435261",   "RetryTimeout": "5000, 5000, 5000",   "ResponseTimeout": "180000, 5000, 5000",   "Amount": "110000" }</pre>
Response:	<pre>{   "Amount": "000000110000",   "ApprovalCode": "",   "CardNo": "621986*****386",   "Merchant": "M00000000000152",   "OptionalField": "",   "OrderId": "435261",   "PacketType": null,   "PcPosStatus": "ارتباط موفق با دستگاه",   "ProcessingCode": "270000",   "ResponseCode": "00",   "Rrn": "",   "Terminal": "C0000301",   "TransactionDate": "1396/04/14",   "TransactionNo": "000018",   "TransactionTime": "15:03:21" }</pre>

## Transaction Reports ۵/۱۳

Prototype:	<a href="http://localhost:8000/api/TransactionReport">http://localhost:8000/api/TransactionReport</a>
Function:	عملیات گزارش تراکنش های آخر.
Be Applicable In:	BlueBird {P3500, CT280, MT280}, Bitel {IC3300, IC3100}
Notes:	حداقل و حداکثر مقدار برای Count بین ۰ تا ۱۳ میباشد. مقادیر مورد تایید برای TransactionType: ۰ (ناموفق)، ۱ (موفق) و ۲ (همه) میباشد.
Request:	<a href="http://localhost:8000/api/TransactionReport">http://localhost:8000/api/TransactionReport</a>
HTTP Method:	POST
Body:	<pre>{   "DeviceIp": "172.24.32.212",   "DevicePort": "8888",   "ConnectionType": "Lan",   "DeviceType": "0",   "OrderId": "435261",   "RetryTimeout": "5000, 5000, 5000",   "ResponseTimeout": "18000, 5000, 5000",   "Count": "15",   "TransactionType": "2" }</pre>
Response:	<pre>{   "OrderId": "435261",   "PcPosStatus": "ارتباط موفق با دستگاه",   "PcPosStatusCode": 4,   "Transactions": [     {       "Amount": "000000000000",       "CardNo": "621986*****3864",       "ResponseCode": "19",       "Rrn": "000000000000",       "TransactionNo": "000009",       "TransactionDate": "20171120",       "TransactionTime": "104102",       "TransactionType": "03",       "OrderId": "78965641"     },     {       "Amount": "000000321300",       "CardNo": "621986*****3864",       "ResponseCode": "00",       "Rrn": "320053286287",       "TransactionNo": "000006",       "TransactionDate": "20171118",       "TransactionTime": "114232",       "TransactionType": "03",       "OrderId": "78965642"     },     {       "Amount": "000000321300",       "CardNo": "621986*****3864",       "ResponseCode": "30",       "Rrn": "000000000000",       "TransactionNo": "000005",       "TransactionDate": "20171118",       "TransactionTime": "114232",       "TransactionType": "03",       "OrderId": "78965642"     }   ] }</pre>
مهر کیفیت:	این سند توسط شرکت پرداخت الکترونیک سداد تولید شده است. هرگونه کپی برداری و یا استفاده دیگر از این سند، بدون کسب مجوز کپی از این شرکت ممنوع می باشد.
صفحه 23 از ۳۷	

	<pre>"TransactionTime": "114208", "TransactionType": "03", "OrderId": "78965643" }, { "Amount": "000000000212", "CardNo": "621986*****3864", "ResponseCode": "61", "Rrn": "000000000000", "TransactionNo": "000004", "TransactionDate": "20171118", "TransactionTime": "114114", "TransactionType": "03", "OrderId": "78965644" } ]</pre>
--	---



## Search ۵/۱۴

Prototype:	<a href="http://localhost:8000/api/Search">http://localhost:8000/api/Search</a>
Function:	جستجو در تراکنش های ذخیره شده در دستگاه پوز.
Be Applicable In:	BlueBird {P3500, CT280, MT280}, <del>Bitel {IC3300, IC3100}</del>
Notes:	
Request:	<a href="http://localhost:8000/api/Search">http://localhost:8000/api/Search</a>
HTTP Method:	POST
Body:	<pre>{   "DeviceIp": "172.24.32.212",   "DevicePort": "8888",   "ConnectionType": "Lan",   "DeviceType": "0",   "OrderId": "435261",   "RetryTimeout": "5000, 5000, 5000",   "ResponseTimeout": "180000, 5000, 5000",   "OrderId": "78965641" }</pre>
Response:	<pre>{   "OrderId": "435261",   "PcPosStatus": "ارتباط موفق با دستگاه",   "PcPosStatusCode": 4,   "Transactions": [     {       "Amount": "000000000000",       "CardNo": "621986*****3864",       "ResponseCode": "19",       "Rrn": "000000000000",       "TransactionNo": "000009",       "TransactionDate": "20171120",       "TransactionTime": "104102",       "TransactionType": "03",       "OrderId": "78965641"     }   ] }</pre>

## Get Card Info ۵/۱۵

Prototype:	<a href="http://localhost:8000/api/CardInfo">http://localhost:8000/api/CardInfo</a>
Function:	عملیات درخواست مشخصات کارت.
Be Applicable In:	BlueBird {P3500, CT280, MT280}, Bitel {IC3300, IC3100}
Notes:	<p>بعد از انجام موفق این متد باید متد خرید عادی را به دستگاه پوز فرستاد.</p> <p>این متد برای فروشگاه های خاص مثل افق کوروش میباشد.</p> <p>روال کار برای گرفتن اطلاعات کارت به صورت مداوم به این صورت است که ابتدا باید در قسمت <b>script</b> صفحه وب یک حلقه ایجاد کرده و این متد را فراخوانی کنید؛ توجه کنید در این زمان دوباره نباید درخواست دریافت اطلاعات کارت به دستگاه فرستاده شود یعنی حلقه برای مدت نهایت ۳ دقیقه در حالت انتظار قرار گیرد. در صورت دریافت اطلاعات کارت و موفق بودن عملیات، اطلاعات تراکنش (مثلا خرید) را بفرستید. بعد از انجام عملیات دوباره حلقه شروع به کار کرده و همین مراحل طی شود.</p>
Request:	<a href="http://localhost:8000/api/CardInfo">http://localhost:8000/api/CardInfo</a>
HTTP Method:	POST
Body:	<pre>{   "DeviceIp": "172.24.32.212",   "DevicePort": "8888",   "ConnectionType": "Lan",   "DeviceType": "0",   "OrderId": "435261",   "RetryTimeout": "5000, 5000, 5000",   "ResponseTimeout": "180000, 5000, 5000", }</pre>
Response:	<pre>{   "CardInfo":     "5bbb7b6569fd751e0b7d31aaf7c0b462728fbf82f111c54c857d11447d63cb6f",   "CardNo": "621986**3864",   "MerchantId": "000000130284816",   "OrderId": "435261",   "PcPosStatus": "ارتباط موفق با دستگاه",   "PcPosStatusCode": 4,   "TerminalId": "2334859", }</pre>

## Get Accounts ۵/۱۶

Prototype:	<a href="http://localhost:8000/api/GetAccounts">http://localhost:8000/api/GetAccounts</a>
Function:	گزارش لیست حسابهای متصل به دستگاه پوز.
Be Applicable In:	BlueBird {P3500, CT280, MT280}
Notes:	
Request:	<a href="http://localhost:8000/api/GetAccounts">http://localhost:8000/api/GetAccounts</a>
HTTP Method:	POST
Body:	<pre>{   "DeviceIp": "172.24.32.212",   "DevicePort": "8888",   "ConnectionType": "Lan",   "DeviceType": "0",   "OrderId": "435261",   "RetryTimeout": "5000, 5000, 5000",   "ResponseTimeout": "180000, 5000, 5000" }</pre>
Response:	<pre>{   "ResponseCode": null,   "ResponseCodeMessage": "",   "Accounts": [     {       "Iban": "IR350170000000106928854003",       "RowNo": "1"     },     {       "Iban": "IR350170000000106928854006",       "RowNo": "4"     },     {       "Iban": "IR350170000000106928854005",       "RowNo": "3"     },     {       "Iban": "IR130170000000123452345008",       "RowNo": "7"     }   ],   "OrderId": "435261",   "PcPosStatus": "ارتباط موفق با دستگاه",   "PcPosStatusCode": 4 }</pre>

## ۵/۱۷ شروع از نو سرویس REST

Prototype:	<a href="http://localhost:8000/api/Restart">http://localhost:8000/api/Restart</a>
Function:	شروع از نو سرویس REST.
Be Applicable In:	BlueBird {P3500, CT280, MT280}, Bitel {IC3300, IC3100}
Notes:	این متد برای شروع از نو سرویس REST از طریق وب میباشد. اگر مشکلی در رابط برنامه نویسی وجود داشته باشد با این متد میتوان سرویس REST را شروع به کار دوباره کرد.
Request:	<a href="http://localhost:8000/api/Restart">http://localhost:8000/api/Restart</a>
HTTP Method:	GET
Body:	null
Response:	null

## ۵/۱۸ :Magic Inquiry

Prototype:	<a href="http://localhost:8000/api/MagicInquiry">http://localhost:8000/api/MagicInquiry</a>
Function:	عملیات استعلام بر اساس شماره مرجع برای دستگاه های Magic.
Be Applicable In:	Magic { X8, X5 }
Notes:	دستگاه POS باید از طریق پورت سریال به رایانه مورد نظر متصل باشد. TerminalId باید به صورت کد معادل تعریف شده در دستگاه Magic مقدار دهی شود. برای مثال "۰۰۱". SerialPort باید به فرمت "COMx" مقدار دهی شود، که X مشخص کننده شماره پورت سریال می باشد. برای مثال پورت شماره ۱ برابر با "COM1" است.
Request:	<a href="http://localhost:8000/api/MagicInquiry">http://localhost:8000/api/MagicInquiry</a>
HTTP Method:	POST
Body:	{ "TerminalId" : "001", "MerchantId" : "011900133", "SerialPort" : "COM1", "RRN" : "144700170050" }
Response:	{ "Amount": "80000", "ApprovalCode": "", "CardNo": "589210*****1", "MerchantId": "011900133", "OptionalField": "000000000000000000", }

	<pre>"OrderId": null, "PacketType": "", "PcPosStatusCode": 4, "ProcessingCode": "", "ResponseCode": "0", "Rrn": "144700170050", "SaleId": "", "TerminalId": "37914470", "TransactionDate": "1398/12/03", "TransactionNo": "531870", "TransactionTime": "09:10:10" }</pre>
--	---

## ۵/۱۹ جواب تراکنش

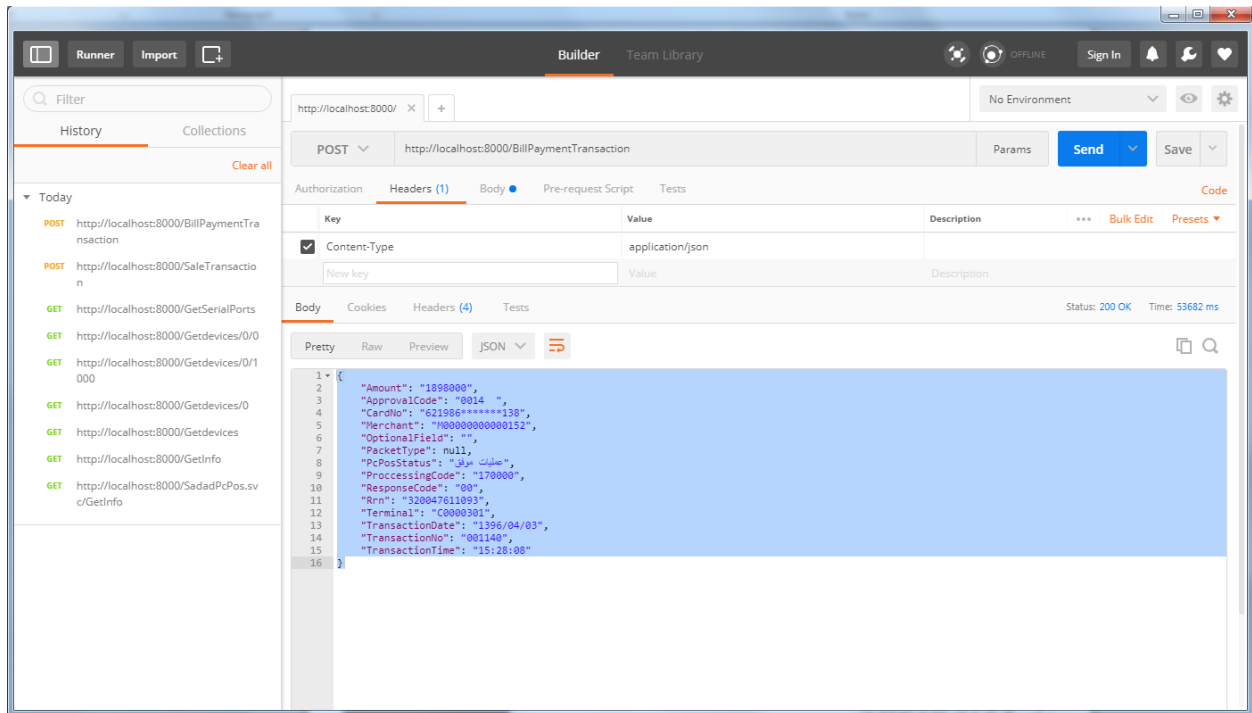
Prototype:	TransactionResponse		
Function:	نتیجه ی عمده تراکنش ها در خصوصیات مورد انتظار آن در چنین کلاسی نمایش داده می شود.		
Property:	Amount	String	مبلغ تراکنش
Property:	ApprovalCode	String	کد تأییدیه تراکنش
Property:	CardNo	String	شماره کارت تراکنش
Property:	MerchantId	String	شماره/کد پذیرنده دستگاه پوز
Property:	OptionalField	String	مقادیر اختیاری <sup>۱</sup>
Property:	OrderId	string	شناسه تراکنش
Property:	PacketType	String	نوع بسته های ارسالی و دریافتی به/از دستگاه پوز <sup>۲</sup>
Property:	PcPosStatus	String	متن پیام کدهای کنترلی ارتباط بین رایانه و دستگاه پوز
Property:	PcPosStatusCode	int	کدهای کنترلی ارتباط بین رایانه و دستگاه پوز <sup>۳</sup>
Property:	ProcessingCode	String	کد پردازشی تراکنش
Property:	ResponseCode	String	کد پاسخ تراکنش <sup>۴</sup>
Property:	ResponseCodeMessage	String	پیام کد پاسخ تراکنش
Property:	Rrn	String	شماره مرجع تراکنش <sup>۵</sup>
Property:	SaleId	String	شناسه خرید
Property:	TerminalId	String	شماره/کد پایانه دستگاه پوز
Property:	TransactionDate	String	تاریخ انجام تراکنش
Property:	TransactionNo	String	شماره پیگیری تراکنش <sup>۶</sup>
Property:	TransactionTime	String	زمان انجام تراکنش
Be Applicable In:	All		
Notes:	<p>۱. برخی پیام های دریافت شده از دستگاه پوز و یا درگاه پرداخت در این خصوصیت برگشت داده می شود.</p> <p>۲. برای توسعه دهندگان نهایی کاربرد ندارد.</p> <p>۳. این کدها برای کنترل ارسال دریافت اطلاعات از رایانه به پوز و بالعکس میباشد:</p> <p>a. ۰: پاسخی دریافت نشد</p> <p>b. ۱: خطا حداکثر دفعات تکرار</p> <p>c. ۳: لغو تراکنش توسط کاربر پوز</p> <p>d. ۲: لغو تراکنش توسط پی سی پوز</p> <p>e. ۴: ارتباط موفق با دستگاه</p> <p>f. ۵: خطای داخلی</p> <p>g. ۶: خطای درخواست همزمان</p> <p>h. در غیر اینصورت: خطای ناشناخته</p> <p>۴. شامل کدهایی است که برای جواب تراکنش استفاده شده است. این جواب ها از درگاه پرداخت یا دستگاه پوز دریافت میشوند که در سند لیست خطاها توضیحات هر کد خطا آمده است.</p>		

	<p>۵. تراکنشی که کد مرجع نداشته باشد حتما مشکلی داشته است و انجام نشده است.</p> <p>۶. شماره ای که به صورت شمارشی اضافه می شود. در صورت بروزرسانی دستگاه و رسیدن به رقم ۹۹۹۹۹۹، صفر می شود.</p>
--	--

## ۶ توضیحات، نکات و تنظیمات

### ۶/۱ تست نرم افزار با REST Client

نرم افزارهای متفاوتی از جمله مرورگرها، افزونه های مرورگر و نرم افزارهای اختصاصی مثل CURL برای تست سرویس های REST وجود دارد. برای تست این محصول تیم PCPOS از نرم افزار Postman استفاده شده است که هم افزونه در مرورگر Chrome دارد هم نرم افزار کاربردی تحت ویندوز. با رفتن به آدرس <https://www.getpostman.com/apps> و دانلود و نصب نرم افزار در بستر مورد نظر، میتوان تست سرویس ذکر شده را انجام داد.



توجه داشته باشید در قسمت Headers مقدار Content-Type = application/json باشد.



## ۶/۲ تنظیمات درخواست در ajax

در زمان استفاده از jQuery باید پارامتر **contentType** مقدار **application/x-www-form-urlencoded** باشد.

به نمونه کد زیر دقت کنید:

```
$.ajax({  
    type: "post",  
    url: "http://localhost:8000/api/Sale",  
    data: JSON.stringify(reqParams),  
    contentType: "application/x-www-form-urlencoded",  
    dataType: "json",  
    crossDomain: true,  
    timeout: 120 * 1000,  
    success: ajaxSuccess,  
    failure: function (errMsg){  
        alert(errMsg);  
    }  
});
```

### ۶/۳ تنظیمات مربوط به پوزهای با قابلیت ارتباط سریال

برای دستگاه هایی که ارتباط آنها با رایانه به صورت کابل سریال می باشد تنظیمات در فایل config سرویس برای راحتی کار برنامه نویسان در نظر گرفته شده است. این تنظیمات توسط پشتیبان شرکت سداد، برنامه نویسان، و کاربران سرویس با در نظر گرفتن اشراف کاربر به تنظیم صحیح، قابل تغییر است. تنظیمات مورد استفاده در زیرمجموعه pcPosDevices در فایل config در دسترس می باشد. این مجموعه به عنوان یک آرایه در نظر گرفته شده است و به ازاء هر پورت COM یا Serial که یک دستگاه به آن متصل می شود، توسط کاربر یک سطر با در نظر گرفتن ورود صحیح خصوصیات هر دستگاه اضافه می شود. هر یک از المان های این مجموعه خصوصیات deviceType, serialPort, terminalId, merchantId, و tag می باشد. لازم به ذکر است این تنظیمات برای شرکت های توسعه نرم افزار در نظر گرفته شده است و می تواند روند توسعه را تسهیل دهد. در زیر نمونه ای از این تنظیمات رو مشاهده می کنید که تنظیمات دو دستگاه متصل به رایانه را دارد:

```
<pcPosDevices>
  <add deviceType="0" serialPort="COM8" terminalId="001" merchantId="01010101010"
tag="pos1" />
  <add deviceType="0" serialPort="COM9" terminalId="001" merchantId="65463102361"
tag="pos2" />
</pcPosDevices>
```

## ۶/۴ اجازه استفاده گواهینامه امنیتی در ویندوز

در صورتی که سرویس پی‌سی‌پوز با پروتکل HTTPS به دلیل عدم وجود گواهینامه امنیتی غیر قابل دسترسی است باید موارد زیر را بررسی شود:

۱. به محل ذخیره سازی گواهینامه های امنیتی ویندوز رفته و بررسی شود آیا گواهینامه امنیتی مربوط به سرویس پی‌سی‌پوز وجود دارد یا نه.

برای دسترسی به محل ذخیره سازی گواهینامه ها موارد زیر را انجام دهید:

- دکمه های Start + R را زده و عبارت mmc را تایپ و Enter کنید
- در پنجره باز شده از منوی File گزینه Add/Remove Snapp-in... را انتخاب کنید
- در لیست سمت چپ گزینه Certificates را انتخاب و دکمه Add را بزنید
- در پنجره باز شده گزینه Computer account را انتخاب و Next را بزنید
- در صفحه بعد گزینه Local computer: (the computer this console is running on) را انتخاب کرده و Finish را بزنید

• در انتها گزینه OK را بزنید تا کارهای مربوط به بررسی یا اضافه کردن گواهینامه را بتوانید انجام دهید.

- به مسیرهای Personal\Certificates و Trusted Root Certification Authorities\Certificates رفته و بررسی کنید آیا گواهینامه با خصوصیات زیر وجود دارد یا نه:

Issued To = localhost

Issued By = localhost

Issuer = "CN=localhost, OU=IT, O=Sadad PSP, L=Tehran, S=TH, C=IR"

Friendly Name = Sadad PcPos REST Service Certificate

در صورت عدم وجود گواهینامه، میتوان یک گواهینامه با ابزارهای موجود با خصوصیات زیر تولید کرده و در محل ذخیره سازی گواهینامه های امنیتی ویندوز آن را ذخیره و استفاده کرد.

DistinguishedName = "CN=localhost, OU=IT, O=Sadad PSP, L=Tehran, S=TH, C=IR";

FriendlyName = "Sadad PcPos REST Service Certificate";

Domains = { "localhost", "sadadpcpos.local", "pcpos.local" };

ProviderType = RSA;

۲. برای ورود گواهینامه موارد زیر را انجام دهید:

- دکمه های Start + R را زده و عبارت mmc را تایپ و Enter کنید
- در پنجره باز شده از منوی File گزینه Add/Remove Snapp-in... را انتخاب کنید
- در لیست سمت چپ گزینه Certificates را انتخاب و دکمه Add را بزنید
- در پنجره باز شده گزینه Computer account را انتخاب و Next را بزنید
- در صفحه بعد گزینه Local computer: (the computer this console is running on) را انتخاب کرده و Finish را بزنید

• در انتها گزینه OK را بزنید تا کارهای مربوط به بررسی یا اضافه کردن گواهینامه را بتوانید انجام دهید.

- به مسیرهای Personal\Certificates و Trusted Root Certification Authorities\Certificates رفته و گزینه Import را بزنید. (در هر دو مسیر باید این کار را انجام دهید)
- در پنجره باز شده گزینه Next را بزنید.
- در صفحه بعد دکمه Browse را زده و به مسیر نصب یا مسیری که فایل گواهینامه وجود دارد رفته و Open را بزنید.
- در ادامه دکمه های Next را زده تا به دکمه Finish برسید و مراحل را تمام کنید.

## ۶/۵ دیواره آتش

ممکن است در سیستم عامل هایی که دیواره آتش<sup>۶</sup> آنها فعال است سرویس REST به درستی کار نکند (برای مثال لیست پایانه‌های تحت شبکه را جستجو و نمایش ندهد. برای حل این مورد باید به تنظیمات دیواره آتش سیستم عامل رفته و اجازه دسترسی مورد نیاز نرم افزارهای استفاده کننده از رابط برنامه نویسی PcPos (در سرویس REST، نرم افزار Sadad.PcPos.WinService.exe) را به آنها دهید. برای انجام این کار میتوانید طبق زیر عمل کنید:

- دکمه Start را زده و وارد Control Panel شوید.
- Windows Firewall را انتخاب کنید و در آنجا گزینه Advanced Settings را در سمت چپ انتخاب کنید.
- در لیست سمت چپ پنجره باز شده گزینه Inbound Rules را انتخاب کنید.
- در لیست سمت راست گزینه New Rule را انتخاب کنید.
- در پنجره باز شده گزینه Program را انتخاب و به مرحله بعد بروید.
- گزینه This program path را انتخاب و دکمه Browse را بزنید.
- در پنجره باز شده به مسیر نصب رفته و فایل اجرایی نرم افزار (Sadad.PcPos.WinService.exe) را انتخاب کنید به

عنوان مثال مسیر زیر:

%ProgramFiles%(x86)\Sadad\PcPos v5.1.764\Service\Sadad.PcPos.WinService.exe

- بعد از تایید به مرحله بعد رفته و گزینه Allow the connection را انتخاب کرده و به مرحله بعد بروید.
- در این قسمت گزینه ها را انتخاب و به مرحله بعد بروید.
- یک نام برای این Rule در نظر بگیرید. مثلا PcPos Sadad.
- دکمه Finish را بزنید.

همین کار را برای Outbound Rules انجام دهید.

<sup>6</sup> Firewall