

بسمه تعالی

راهنمای نحوه اتصال به درگاه پرداخت اینترنتی

نسخه : V_1.009.980717

سنزپی



SIZPay.com



تغییرات	نسخه
تهیه و تدوین	V_1.001.960429
بازبینی و اصلاح	V_1.002.980218
اصلاح آدرس های اینترنتی	V_1.003.980229
- تغییر پارامترهای ورودی متد دریافت توکن - افزودن بخش گزارش (متد گزارش اطلاعات پرداخت با شناسه پرداخت)	V_1.004.980416
- افزودن InvoiceNo به مقادیر بازگشتی از متد های بند های ۳ و ۴ و ۵ (متدهای سرویس پرداخت) - اصلاح مقادیر بازگشتی از متد گزارش اطلاعات پرداخت با شناسه پرداخت	V_1.005.980417
- بازبینی کامل متد ها - افزودن شمای پیاده سازی هر متد - افزودن بخش وب سرویس Soap - افزودن بخش اتصال به صفحه درگاه پرداخت شخصی به صورت مستقیم از App پذیرنده	V_1.006.980508
افزودن AppExtraInf به متد های Confirm , Reverse و نتیجه تراکنش در هر دو وب سرویس Soap و Rest	V_1.007.980509
افزودن توکن به مقادیر بازگشتی از کلیه متدهای پرداخت	V_1.008.980516
حذف AppExtraInf از SignData در متد دریافت توکن	V_1.009.980717



فهرست

۴	مقدمه
۴	پیش نیازها
۵	شرح روال پرداخت
۶	شرح سرویس های پرداخت
۶	وب سرویس Rest (Web API)
۶	متد دریافت توکن (GetToken)
۶	• ورودی متد
۷	• خروجی متد
۷	• آماده سازی رشته SignData
۸	• روش رمز کردن رشته
۹	• نمونه کد رمزنگاری AES به زبان C#
۱۰	• نمونه کد رمزنگاری AES به زبان PHP
۱۱	• نمونه کد رمزنگاری SHA256 به زبان C#
۱۲	راهنمای پذیرنده (Redirect) به صفحه درگاه پرداخت اینترنتی
۱۲	• آماده سازی رشته SignData
۱۳	• روش رمز کردن رشته
۱۴	ارسال نتیجه تراکنش به آدرس بازگشت پذیرنده
۱۵	متد تایید پرداخت (Confirm)
۱۵	• ورودی متد
۱۵	• خروجی متد
۱۶	• آماده سازی رشته SignData
۱۷	• روش رمز کردن رشته
۱۸	متد بازگشت پرداخت Reverse
۱۸	• ورودی متد
۱۸	• خروجی متد
۱۹	• آماده سازی رشته SignData
۲۰	• روش رمز کردن رشته
۲۰	سرویس های گزارش
۲۰	متد گزارش اطلاعات پرداخت با شناسه پرداخت
۲۰	• ورودی متد



راهنمای نحوه اتصال به درگاه پرداخت اینترنتی - سیز پی

- ۲۰..... خروجی متد •
- ۲۱..... آماده سازی رشته SignData •
- ۲۲..... روش رمز کردن رشته •
- ۲۳..... وب سرویس Soap (Simple Object Access Protocol) •
- ۲۳..... متد دریافت توکن (GetToken) •
- ۲۳..... ورودی متد •
- ۲۵..... خروجی متد •
- ۲۵..... راهنمای پذیرنده (Redirect) به صفحه درگاه پرداخت اینترنتی •
- ۲۶..... ارسال نتیجه تراکنش به آدرس بازگشت پذیرنده •
- ۲۷..... متد تایید پرداخت (Confirm) •
- ۲۷..... ورودی متد •
- ۲۸..... خروجی متد •
- ۲۹..... متد بازگشت پرداخت (Reverse) •
- ۲۹..... ورودی متد •
- ۳۰..... خروجی متد •
- ۳۱..... شرح سرویس های گزارش •
- ۳۱..... متد گزارش اطلاعات پرداخت با شناسه پرداخت •
- ۳۱..... ورودی متد •
- ۳۱..... خروجی متد •
- ۳۳..... اتصال به صفحه درگاه پرداخت شخصی به صورت مستقیم از App پذیرنده •



مقدمه

برای راحتی کاربران و پرهیز از مشکلات ناشی از برخی راه حل های دشوار، که پیاده سازی درگاه پرداخت اینترنتی را برای پذیرندگان اینترنتی دشوار می سازد، در این مستند از فناوری های وب سرویس Rest (Web API) و وب سرویس Soap برای برقراری خدمات پرداخت استفاده نموده است.

پیش نیازها

برای استفاده از این سرویس و اتصال به سرور پرداخت، لازم است ابتدا از آدرس زیر اقدام به ثبت نام نمایید :

<https://www.sizpay.com/>

در صورتی که درخواست مزبور مورد موافقت قرار بگیرد، پذیرنده از سوی سیزپی پنج نوع اطلاعات زیر را دریافت خواهد نمود:

❖ کد پذیرنده (Merchant Code)

❖ کد ترمینال (Terminal Code)

❖ کلید رمزنگاری (KEY) (کلید ۱)

❖ IV کلید رمزنگاری (کلید ۲)

❖ کلید امضای الکترونیکی

لازم است اطمینان حاصل کنید که پورت های ۴۴۳ و ۸۰ روی سرور پذیرنده باز هستند و می توانند روی این دو پورت اطلاعات را ارسال و دریافت نمایند.

جهت تست درگاه پرداخت می توانید از آدرس زیر استفاده نمایید:

<https://doc.sizpay.com/>



شرح روال پرداخت

۱- دریافت توکن

پس از تشکیل سبد خرید و انتخاب گزینه پرداخت توسط دارنده کارت، پذیرنده جهت شروع تراکنش اقدام به دریافت توکن پرداخت می نماید (رجوع به بخش سرویس دریافت توکن). این توکن یکبار مصرف و به مدت محدودی قابل استفاده می باشد.

۲- راهنمای دارنده کارت به صفحه پرداخت

پس دریافت توکن سایت پذیرنده دارنده کارت را به صفحه درگاه پرداخت راهنمایی (Redirect) می نماید.

۳- تکمیل فرآیند خرید با ارسال تاییده یا تراکنش بازگشت

صفحه درگاه پرداخت اینترنتی پس از تکمیل و تایید پرداخت دارنده در صورت موفق بودن پرداخت رسید تراکنش را نمایش داده و به آدرس تعیین شده توسط پذیرنده (ReturnURL) بازگشت داده می شود (با استفاده از متد POST). سایت پذیرنده با بررسی نتیجه تراکنش اقدام به ارسال تاییدیه یا بازگشت تراکنش می نماید.



شرح سرویس های پرداخت

وب سرویس Rest (Web API)

در بخش زیر به شرح جزئیات متدهای مورد استفاده فرآیند خرید پرداخته می شود. تمامی متدهای این فصل به صورت Web API توسعه داده شده اند.

متد دریافت توکن (GetToken)

این متد در شروع فرآیند پرداخت جهت دریافت توکن مورد استفاده قرار می گیرد.
آدرس متد به شرح ذیل می باشد:

<https://rt.sizpay.com/api/Payment/GetToken>

ورودی ها و خروجی های این متد به شرح ذیل است:

ورودی متد

نشانه	نام پارامتر	مفهوم	مثال
P1	MerchantID	شماره پذیرنده	
P2	TerminalID	شماره ترمینال	
P3	Amount	مبلغ فاکتور به ریال بدون هیچ Separator ای می بایست ارسال شود.	۱۰۰۰
P4	DocDate	تاریخ شمسی فاکتور با فرمت ۱۳۰۰/۱۲/۲۹ هر فرمتی به جز فرمت ذکر شده قابل قبول نمی باشد. صحیح بودن تاریخ بررسی می شود	۱۳۹۵/۱۱/۲۲
P5	OrderID	شناسه خرید. لازم است پذیرنده در هنگام ارسال درخواست خرید خود، شناسه منحصر به فرد را ارسال نماید. این شناسه باید در هر روز به صورت یکتا باشد (ترجیحا برای پیگیری های بهتر همیشه یکتا و منحصر به فرد باشد)	
P6	ReturnURL	در این فیلد لازم است آدرس Callback ای که پاسخ درگاه پرداخت به آن صفحه ارسال خواهد شد را ارسال کنید. این	



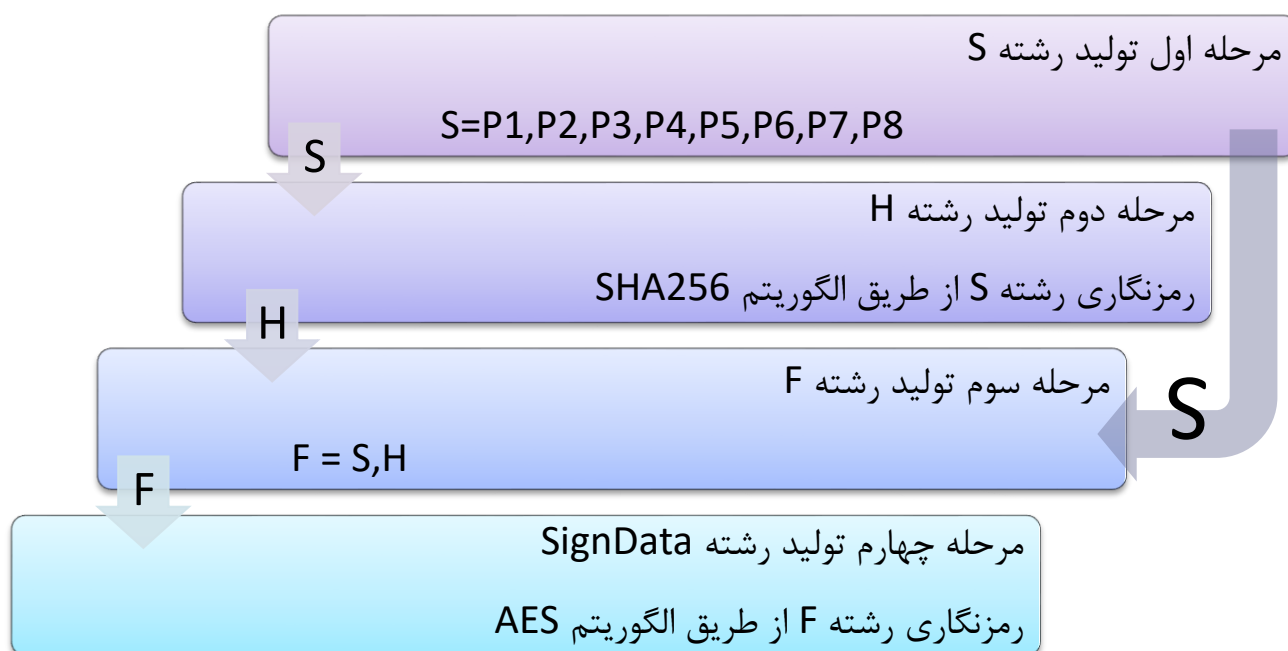
راهنمای نحوه اتصال به درگاه پرداخت اینترنتی - سیز پی

	آدرس می بایست با HTTP:// یا HTTPS:// شروع شود(صحت آدرس چک می شود) و اکیدا توصیه می شود از شروع کردن URL ها با آدرس IP اجتناب بفرمائید.		
P7	اطلاعات اضافی، به صورت رشته می باشد (حداکثر تا ۱۰۰ کاراکتر)	ExtraInf	
P8	شناسه پرداخت. به صورت رشته ای کاملا عددی و حداکثر طول ۵۰ می باشد.	InvoiceNo	
P9	اطلاعات اضافی مورد نیاز برنامه مانند نام پرداخت کننده، می JSON شماره همراه پرداخت کننده و ... که با فرمت باشد. <pre>{ "PayerNm": "نام پرداخت کننده", "PayerMobile": "موبایل", "PayerEmail": "ایمیل", "Descr": "توضیحات", "PayerIP": "آی پی دستگاه پرداخت کننده", "PayTitle": "عنوان پرداخت" }</pre> لازم به ذکر است حتما باید فرمت زیر رعایت شود. در صورتی که هریک از موارد بالا مورد نیاز نمی باشد رشته تهی ارسال نمایید.	AppExtraInf	
H	اطلاعات تراکنش به صورت رمزنگاری شده (نحوه تولید در ادامه آمده است)	SignData	

خروجی متد

نام پارامتر	مفهوم
ResCod	کد نتیجه متد، در صورت ۰ یا ۰۰ نتیجه موفق و در غیر اینصورت نا موفق می باشد
Message	شرح نتیجه متد
Token	توکن، در صورت موفقیت تراکنش رشته توکن بازگشت داده می شود. توکن ارسالی یکبار مصرف بوده و به مدت محدود (حداکثر ۱۰ دقیقه) قابل استفاده می باشد

آماده سازی رشته SignData



برای آماده سازی این رشته لازم است نخست رشته ای ایجاد کنید از ۹ پارامتر زیر که صرفاً با یک کاراکتر کامای لاتین (,) از یکدیگر جدا شوند. توجه کنید که Delimiter تنها کاراکتر کاماست و استفاده از سایر کاراکترها همانند فاصله و غیره برای جداسازی پارامترها مجاز نیست.

این ۹ پارامتر می بایست صرفاً با یک ترتیب مشخص در رشته قرار بگیرند و پذیرنده مجاز به تغییر محل نسبی قرارگیری پارامتری در درون رشته نمی باشد(از چپ به راست).

S=P1,P2,P3,P4,P5,P6,P7,P8

نکته مهم: چنانچه به اقتضا شرایط بخواهید فیلدی را خالی بگذارید لازم است در هر حال و هر شرایطی کاراکتر Delimiter را درج نمایید.

روش رمز کردن رشته

برای رمز نگاری رشته بدست آمده از مراحل بالا و همچنین رشته های رمز شده ای که در متدهای دیگر درگاه پرداخت مورد نیاز هستند لازم است از الگوریتم AES و SHA2 با مشخصه های زیر استفاده شود:

• AES

❖ اندازه کلید و بلوک: ۲۵۶ بیت

❖ نوع Padding: PKCS7

❖ کلید رمزنگاری: همان کلیدی که از سوی درگاه پرداخت به پذیرنده تخصیص داده شده است.

❖ IV: همان IV ای که از سوی الکترونیک درگاه پرداخت به پذیرنده تخصیص داده شده است.

❖ Encoding مورد استفاده شده: UTF-8

• SHA256



❖ Encoding مورد استفاده شده: UTF-8

❖ Hash-based Message Authentication Code (HMAC): SHA256

❖ کلید امضای الکترونیکی: همان کلیدی که از سوی درگاه پرداخت به پذیرنده تخصیص داده شده است .

روش رمز کردن به صورت زیر می باشد:

۱- ابتدا رشته بدست آمده (S) را با الگوریتم SHA256 رمز نگاری نمایید (H)

۲- سپس کد رمز نگاری شده مرحله قبل (H) را با جدا کننده کاما لاتین (,) به انتهای رشته (S) اضافه نمایید (F)

$$F = S, H$$

۳- در انتها عبارت بدست آمده (F) را با الگوریتم AES رمز نگاری نمایید

در زیر نمونه کدهایی برای رمزنگاری به روش بالا در زبان های مختلف آورده شده است:

🚩 نمونه کد رمزنگاری AES به زبان C#

```
public class AES
{
    string AES_Key = string.Empty;
    string AES_IV = string.Empty;
    public AES(string AES_Key, string AES_IV)
    {
        this.AES_Key = AES_Key;
        this.AES_IV = AES_IV;
    }
    public bool Encrypt(String Input, out string encryptedString)
    {
        try
        {
            var aes = new RijndaelManaged();
            aes.KeySize = 256;
            aes.BlockSize = 128;
            aes.Padding = PaddingMode.PKCS7;
            aes.Key = Convert.FromBase64String(this.AES_Key);
            aes.IV = Convert.FromBase64String(this.AES_IV);
            var encrypt = aes.CreateEncryptor(aes.Key, aes.IV);
            byte[] xBuff = null;
            using (var ms = new MemoryStream())
            {
                using (var cs = new CryptoStream(ms, encrypt, CryptoStreamMode.Write))
                {
                    byte[] xXml = Encoding.UTF8.GetBytes(Input);
                    cs.Write(xXml, 0, xXml.Length);
                }
                xBuff = ms.ToArray();
            }
            encryptedString = Convert.ToBase64String(xBuff);
            return true;
        }
        catch (Exception ex)
        {
            // Handle exception
        }
    }
}
```



```
{
    encryptedString = string.Empty;
    return false;
}
}
public bool Decrypt(String Input, out string decodedString)
{
    try
    {
        RijndaelManaged aes = new RijndaelManaged();
        aes.KeySize = 256;
        aes.BlockSize = 128;
        aes.Mode = CipherMode.CBC;
        aes.Padding = PaddingMode.PKCS7;
        aes.Key = Convert.FromBase64String(this.AES_Key);
        aes.IV = Convert.FromBase64String(this.AES_IV);
        var decrypt = aes.CreateDecryptor();
        byte[] xBuff = null;
        using (var ms = new MemoryStream())
        {
            using (var cs = new CryptoStream(ms, decrypt, CryptoStreamMode.Write))
            {
                byte[] xXml = Convert.FromBase64String(Input);
                cs.Write(xXml, 0, xXml.Length);
            }
            xBuff = ms.ToArray();
        }
        decodedString = Encoding.UTF8.GetString(xBuff);
        return true;
    }
    catch (Exception ex)
    {
        decodedString = string.Empty;
        return false;
    }
}
}
```

نمونه کد رمزنگاری AES به زبان PHP

```
<?php
// enable extension=php_mcrypt.dll AND extension=php_soap.dll on php.ini

date_default_timezone_set('Asia/Tehran');
$KEY = "Your KEY";
$IV = "Your IV";
function addpadding($string, $blocksize = 16)
{
    $len = strlen($string);
    $pad = $blocksize - ($len % $blocksize);
    $string .= str_repeat(chr($pad), $pad);
    return $string;
}
function strippadding($string)
{
    $slast = ord(substr($string, -1));
    $slastc = chr($slast);
```



```
$pcheck = substr($string, -$slast);
if(preg_match("/$slastc{". $slast."/}", $string))
{
    $string = substr($string, 0, strlen($string)-$slast);
    return $string;
}
else
{
    return false;
}
}
function encrypt($string = "")
{
    global $KEY,$IV;
    $key = base64_decode($KEY);
    $iv = base64_decode($IV);
    return base64_encode(mcrypt_encrypt(MCRYPT_RIJNDAEL_128, $key, addpadding($string),
        MCRYPT_MODE_CBC, $iv));
}
function decrypt($string = "")
{
    global $KEY,$IV;
    $key = base64_decode($KEY);
    $iv = base64_decode($IV);
    $string = base64_decode($string);
    return strippadding(mcrypt_decrypt(MCRYPT_RIJNDAEL_128, $key, $string, MCRYPT_MODE_CBC,
        $iv));
}
?>
```

نمونه کد رمزنگاری SHA256 به زبان C#

```
public string pubFunStrGetSHA2(string prmKey, string prmPlainData)
{
    byte[] varArrPlainData;
    byte[] varArrCypherData;
    byte[] varArrSHA2Key;
    try
    {
        varArrSHA2Key = System.Text.Encoding.UTF8.GetBytes(prmKey);
        varArrPlainData = System.Text.Encoding.UTF8.GetBytes(prmPlainData);
        var varSHA256 = new System.Security.Cryptography.HMACSHA256(varArrSHA2Key);
        varArrCypherData = varSHA256.ComputeHash(varArrPlainData);
        return Convert.ToBase64String(varArrCypherData);
    }
    catch (Exception)
    {
        return string.Empty;
    }
}
```



راهنمای نحوه اتصال به درگاه پرداخت اینترنتی - سیز پی

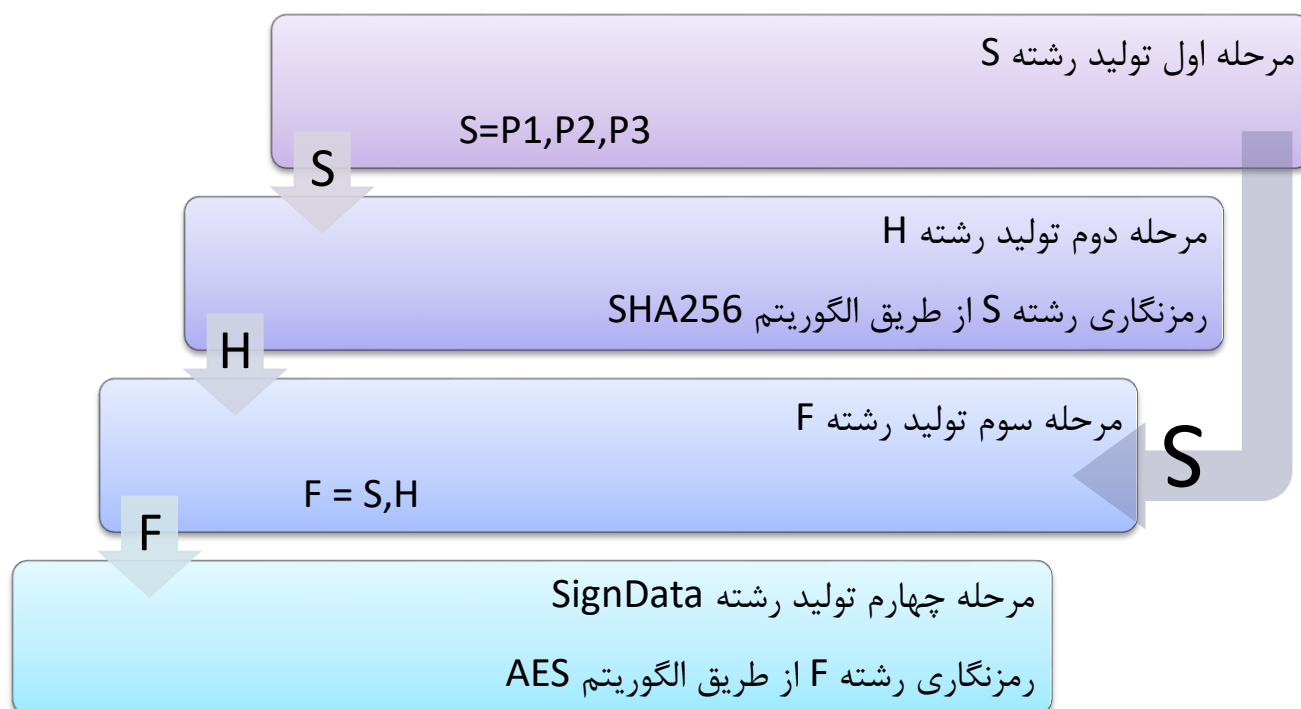
راهنمای پذیرنده (Redirect) به صفحه درگاه پرداخت اینترنتی

برای ارجاع مشتری به درگاه پرداخت، لازم است مقادیر زیر را به صورت POST به آدرس زیر ارسال نمایید با این کار مشتری پذیرنده به درگاه پرداخت وارد شده و آماده انجام تراکنش مالی می گردد.

<https://rt.sizpay.com/Route/Payment>

نشانه	نام پارامتر	مفهوم
P1	MerchantID	شماره پذیرنده
P2	TerminalID	شماره ترمینال
P3	Token	توکن
H	SignData	اطلاعات تراکنش به صورت رمزنگاری شده (نحوه تولید در ادامه آمده است)

آماده سازی رشته SignData



برای آماده سازی این رشته لازم است نخست رشته ای ایجاد کنید از ۳ پارامتر زیر که صرفاً با یک کاراکتر کامای لاتین (,) از یکدیگر جدا شوند. توجه کنید که Delimiter تنها کاراکتر کاماست و استفاده از سایر کاراکترها همانند فاصله و غیره برای جداسازی پارامترها مجاز نیست.

این ۳ پارامتر می بایست صرفاً با یک ترتیب مشخص در رشته قرار بگیرند و پذیرنده مجاز به تغییر محل نسبی قرارگیری پارامتری در درون رشته نمی باشد(از چپ به راست).



S=P1,P2,P3

نکته مهم: چنانچه به اقتضا شرایط بخواهید فیلدی را خالی بگذارید لازم است در هر حال و هر شرایطی کاراکتر Delimiter را درج نمائید.

روش رمز کردن رشته

روش رمز کردن به صورت زیر می باشد:

۱- ابتدا رشته بدست آمده (S) را با الگوریتم SHA256 رمز نگاری نمایید (H)

۲- سپس کد رمز نگاری شده مرحله قبل (H) را با جدا کننده کاما لاتین (,) به انتهای رشته (S)

افزافه نمایید (F) $F=S,H$

۳- در انتها عبارت بدست آمده (F) را با الگوریتم AES رمز نگاری نمایید

برای راحتی کار می توانید از تابع زیر برای انجام عملیات ارجاع به درگاه پرداخت استفاده کنید. مبرهن است که این تابع را باید برای کاربرد خاص خود تغییر دهید تا متناسب با سناریوی مدنظرتان گردد:

```
<script language="javascript" type="text/javascript">
function postData() {
    var form = document.createElement("form");
    form.setAttribute("method", "POST");
    form.setAttribute("action", "https://rt.sizpay.com/Route/Payment");
    form.setAttribute("target", "_self");
    var hiddenField = document.createElement("input");
    hiddenField.setAttribute("name", "MerchantID");
    hiddenField.setAttribute("value", MrchIDValue);
    form.appendChild(hiddenField);
    var hiddenField1 = document.createElement("input");
    hiddenField1.setAttribute("name", "TerminalID");
    hiddenField1.setAttribute("value", TrmnlIDValue);
    form.appendChild(hiddenField1);
    var hiddenField2 = document.createElement("input");
    hiddenField2.setAttribute("name", "Token");
    hiddenField2.setAttribute("value", TokenValue);
    form.appendChild(hiddenField2);
    var hiddenField3 = document.createElement("input");
    hiddenField3.setAttribute("name", "SignData");
    hiddenField3.setAttribute("value", SignDataValue);
    form.appendChild(hiddenField3);
    document.body.appendChild(form);
    form.submit();
    document.body.removeChild(form);
}
</script>
```



ارسال نتیجه تراکنش به آدرس بازگشت پذیرنده

پس از تکمیل فرآیند پرداخت توسط کاربر مقادیر زیر و با متد POST به آدرس بازگشت اعلام شده توسط پذیرنده ارسال می گردد و سامانه پذیرنده می بایست در صورت موفق بودن پرداخت نسبت ارسال تاییدیه یا تراکنش بازگشت اقدام نماید.

نام پارامتر	مفهوم
ResCod	کد نتیجه پرداخت، در صورت ۰ یا ۰۰ نتیجه موفق و در غیر اینصورت نا موفق می باشد
Message	شرح نتیجه پرداخت
MerchantID	شماره پذیرنده
TreminalID	شماره ترمینال
InvoiceNo	شناسه پرداخت
ExtraInf	اطلاعات اضافی
AppExtraInf	اطلاعات اضافی مورد نیاز برنامه مانند نام پرداخت کننده، شماره همراه پرداخت کننده و ... که با فرمت JSON می باشد. <pre>{ "PayerNm": "نام پرداخت کننده", "PayerMobile": "موبایل", "PayerEmail": "ایمیل", "Descr": "توضیحات", "PayerIP": "آی پی دستگاه پرداخت کننده", "PayTitle": "عنوان پرداخت" }</pre>
Token	توکن

پس از دریافت پارامترهای فوق ابتدا باید نتیجه تراکنش (ResCode) چک شود که در صورتیکه 0 یا 00 بود، کار ادامه یابد و در غیراینصورت تراکنش ناموفق بوده و وضعیت آن مختومه می باشد.

نکته مهم : در صورت عدم ارسال تاییدیه یا درخواست بازگشت در مدت تعیین شده تراکنش صورت خودکار بازگشت داده می شود.



متد تایید پرداخت (Confirm)

پس از اطلاع یافتن پذیرنده از موفق بودن تراکنش مالی در درگاه پرداخت، لازم است پذیرنده بلافاصله این وب سرویس متد را فراخوانی نماید بدیهی است در پذیرنده فقط در صورت موفق بودن پرداخت مجاز به فراخوانی این متد می باشد برای تایید پرداخت، لازم است مقادیر زیر را به صورت POST به آدرس زیر ارسال نمایید.

<https://rt.sizpay.com /api/Payment/Confirm>

ورودی متد

نشانه	نام پارامتر	مفهوم
P1	MerchantID	شماره پذیرنده
P2	TerminalID	شماره ترمینال
P3	Token	توکن
H	SignData	اطلاعات تراکنش به صورت رمزنگاری شده

خروجی متد

نام پارامتر	مفهوم
ResCod	کد نتیجه پرداخت، در صورت ۰ یا ۰۰ نتیجه موفق و در غیر اینصورت نا موفق می باشد
Message	شرح نتیجه پرداخت
MerchantID	شماره پذیرنده
TerminalID	شماره ترمینال
OrderID	شناسه خرید
TransNo	شماره تراکنش (اختصاصی)
RefNo	شماره ارجاع
TraceNo	شماره پیگیری
ExtraInf	اطلاعات اضافی
Amount	مبلغ تراکنش
CardNo	شماره کارت به صورت مختصر
TransDate	تاریخ و ساعت میلادی تراکنش



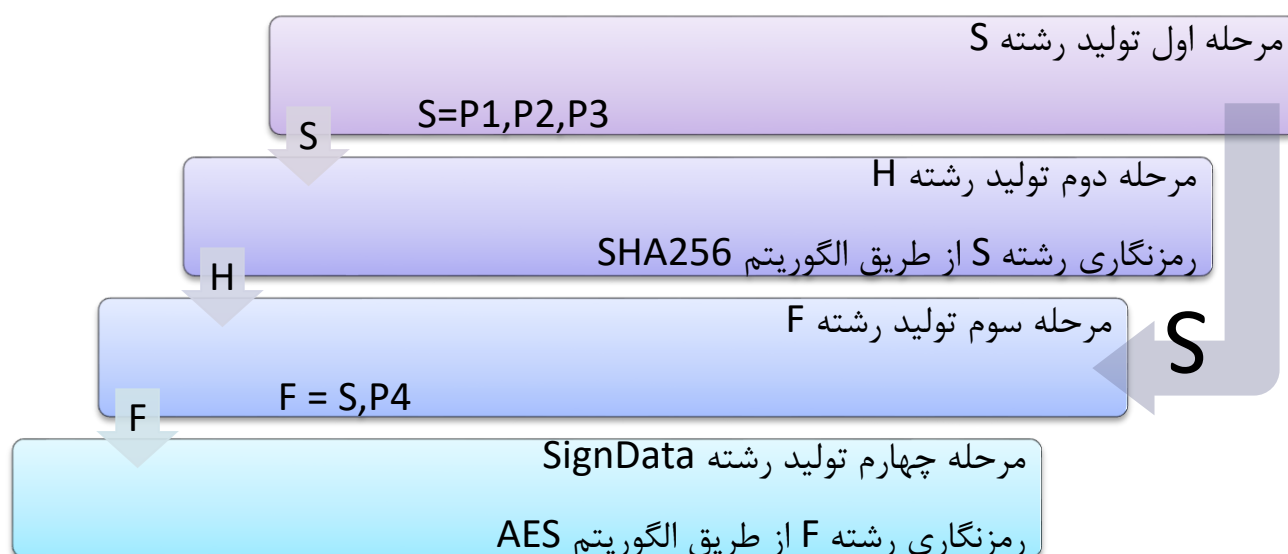
راهنمای نحوه اتصال به درگاه پرداخت اینترنتی - سیز پی

شناسه پرداخت	InvoiceNo
اطلاعات اضافی مورد نیاز برنامه مانند نام پرداخت کننده، شماره همراه پرداخت کننده و ... که با فرمت JSON می باشد.	AppExtraInf
<pre>{ "PayerNm": "نام پرداخت کننده", "PayerMobile": "موبایل", "PayerEmail": "ایمیل", "Descr": "توضیحات", "PayerIP": "آی پی دستگاه پرداخت کننده", "PayTitle": "عنوان پرداخت" }</pre>	
توکن	Token

موارد زیر در رابطه با فراخوانی این متد حائز اهمیت هستند:

- ❖ یک تراکنش را نمی تواند بیش از یکبار Confirm نمود.
- ❖ تراکنشی که برایش درخواست بازگشت مبلغ تراکنش شده را نمی توان Confirm نمود.
- ❖ تراکنش مالی غیرموفق را نمی توان Confirm نمود.
- ❖ عدم ارسال درخواست Confirm تا ۱۰ دقیقه بعد از انجام تراکنش (به وقت سرور پرداخت) موجب خواهد شد تا بصورت خودکار تراکنش مزبور Reverse شود و مبلغ آن به حساب مشتری برگشت داده می شود.
- ❖ همانگونه که از جدول کدهای نتیجه فراخوانی این متد پیداست، نتیجه 0 یا 00 به معنای موفقیت آمیز بودن عملیات است.

آماده سازی رشته SignData





راهنمای نحوه اتصال به درگاه پرداخت اینترنتی - سیز پی

برای آماده سازی این رشته لازم است نخست رشته ای ایجاد کنید از ۳ پارامتر زیر که صرفاً با یک کاراکتر کامای لاتین (,) از یکدیگر جدا شوند. توجه کنید که Delimiter تنها کاراکتر کاماست و استفاده از سایر کاراکترها همانند فاصله و غیره برای جداسازی پارامترها مجاز نیست.

این ۳ پارامتر می بایست صرفاً با یک ترتیب مشخص در رشته قرار بگیرند و پذیرنده مجاز به تغییر محل نسبی قرارگیری پارامتری در درون رشته نمی باشد (از چپ به راست).

$S=P1,P2,P3$

نکته مهم: چنانچه به اقتضا شرایط بخواهید فیلدی را خالی بگذارید لازم است در هر حال و هر شرایطی کاراکتر Delimiter را درج نمائید.

روش رمز کردن رشته

روش رمز کردن به صورت زیر می باشد:

۱- ابتدا رشته بدست آمده (S) را با الگوریتم SHA256 رمز نگاری نمایید (H)

۲- سپس کد رمز نگاری شده مرحله قبل (H) را با جدا کننده کاما لاتین (,) به انتهای رشته (S)

اضافه نمایید (F) $F=S,H$

۳- در انتها عبارت بدست آمده (F) را با الگوریتم AES رمز نگاری نمایید



متد بازگشت پرداخت Reverse

چنانچه پذیرنده بخواهد مبلغ تراکنش مالی موفق را به حساب پذیرنده بازگرداند، به شرط آنکه قبلاً تراکنش مزبور را Confirm ننموده باشد، لازم است از متد Reverse استفاده کند. با فراخوانی این متد، درخواست پذیرنده برای پردازش و عودت وجه در سیکل پردازش سرور پرداخت قرار می گیرد. برای بازگشت پرداخت، لازم است مقادیر زیر را به صورت POST به آدرس زیر ارسال نمایید

<https://rt.sizpay.com /api/Payment/Reverse>

ورودی متد

نشانه	نام پارامتر	مفهوم
P1	MerchantID	شماره پذیرنده
P2	TerminalID	شماره ترمینال
P3	Token	توکن
H	SignData	اطلاعات تراکنش به صورت رمزنگاری شده

خروجی متد

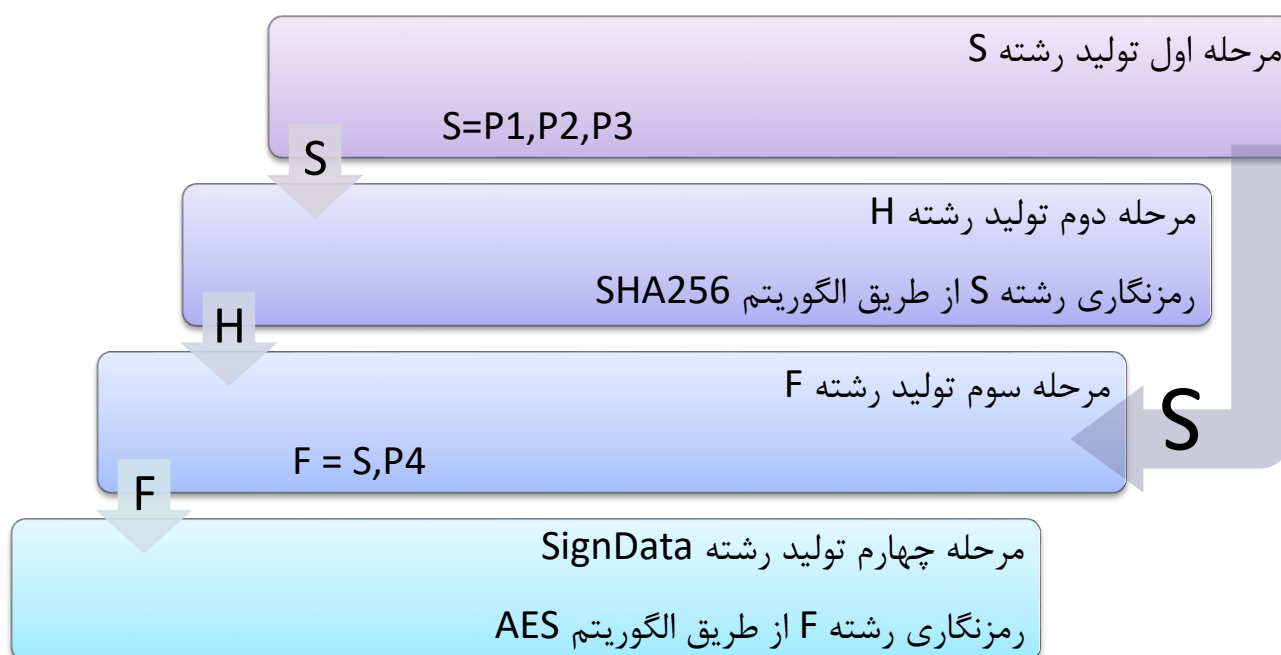
نام پارامتر	مفهوم
ResCod	کد نتیجه پرداخت، در صورت ۰ یا ۰۰ نتیجه موفق و در غیر اینصورت ناموفق می باشد
Message	شرح نتیجه پرداخت
MerchantID	شماره پذیرنده
TerminalID	شماره ترمینال
OrderID	شناسه خرید
TransNo	شماره تراکنش (اختصاصی)
RefNo	شماره ارجاع
TraceNo	شماره پیگیری
ExtraInf	اطلاعات اضافی
Amount	مبلغ تراکنش
CardNo	شماره کارت به صورت مختصر
TransDate	تاریخ و ساعت میلادی تراکنش



راهنمای نحوه اتصال به درگاه پرداخت اینترنتی - سیز پی

شناسه پرداخت	InvoiceNo
اطلاعات اضافی مورد نیاز برنامه مانند نام پرداخت کننده، شماره همراه پرداخت کننده و ... که با فرمت JSON می باشد.	AppExtraInf
<pre>{ "PayerNm": "نام پرداخت کننده", "PayerMobile": "موبایل", "PayerEmail": "ایمیل", "Descr": "توضیحات", "PayerIP": "آی پی دستگاه پرداخت کننده", "PayTitle": "عنوان پرداخت" }</pre>	
توکن	Token

آماده سازی رشته SignData



برای آماده سازی این رشته لازم است نخست رشته ای ایجاد کنید از ۳ پارامتر زیر که صرفاً با یک کاراکتر کامای لاتین (,) از یکدیگر جدا شوند. توجه کنید که Delimiter تنها کاراکتر کاماست و استفاده از سایر کاراکترها همانند فاصله و غیره برای جداسازی پارامترها مجاز نیست.

این ۳ پارامتر می بایست صرفاً با یک ترتیب مشخص در رشته قرار بگیرند و پذیرنده مجاز به تغییر محل نسبی قرارگیری پارامتری در درون رشته نمی باشد (از چپ به راست).

$S = P1, P2, P3$

نکته مهم: چنانچه به اقتضا شرایط بخواهید فیلدی را خالی بگذارید لازم است در هر حال و هر شرایطی کاراکتر Delimiter را درج نمایید.



راهنمای نحوه اتصال به درگاه پرداخت اینترنتی - سیزپی

روش رمز کردن رشته

روش رمز کردن به صورت زیر می باشد:

- ۱- ابتدا رشته بدست آمده (S) را با الگوریتم SHA256 رمز نگاری نمایید (H)
- ۲- سپس کد رمز نگاری شده مرحله قبل (H) را با جدا کننده کاما لاتین (,) به انتهای رشته (S) اضافه نمایید (F)

$$F=S,H$$
- ۳- در انتها عبارت بدست آمده (F) را با الگوریتم AES رمز نگاری نمایید

سرویس های گزارش

متد گزارش اطلاعات پرداخت با شناسه پرداخت

<https://rt.sizpay.com/api/Report/TransactionListByInvoiceNo>

ورودی متد

نشانه	نام پارامتر	مفهوم
P1	MerchantID	شماره پذیرنده
P2	TerminalID	شماره ترمینال
P3	InvoiceNo	شناسه پرداخت
P4	InvoiceDate	تاریخ شمسی فاکتور با فرمت ۱۳۰۰/۱۲/۲۹ هر فرمتی به جز فرمت ذکر شده قابل قبول نمی باشد. صحیح بودن تاریخ بررسی می شود
H	SignData	اطلاعات تراکنش به صورت رمزنگاری شده

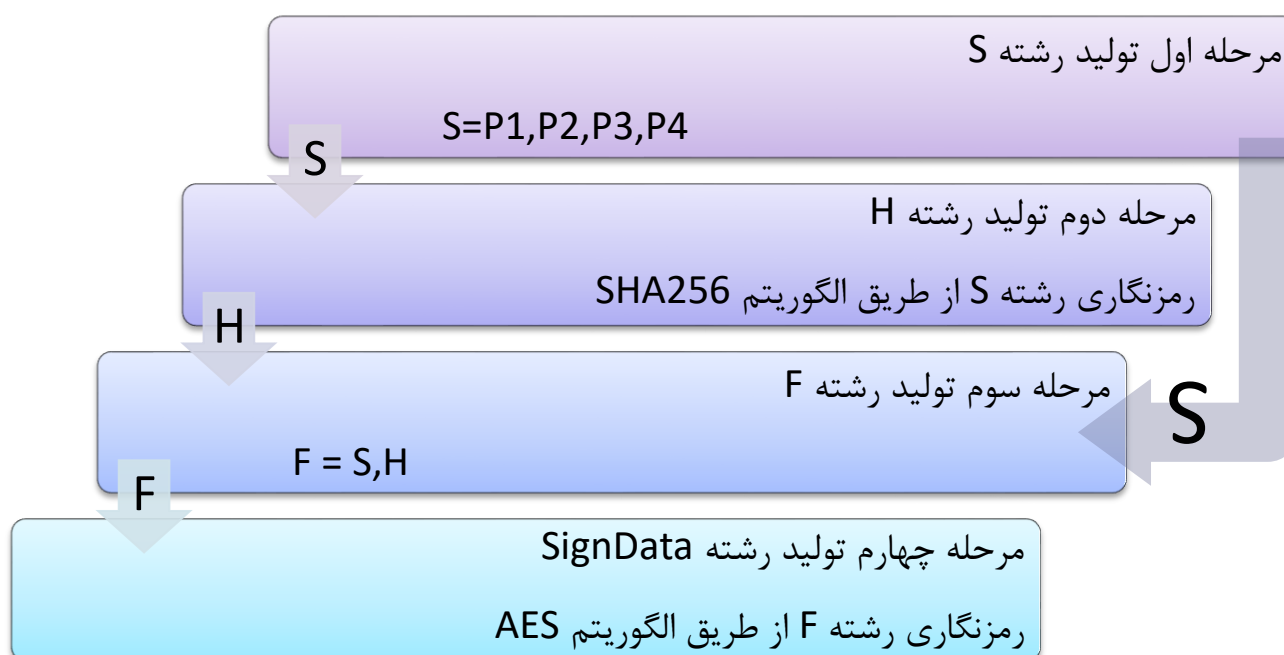
خروجی متد

نام پارامتر	مفهوم
ResCod	کد نتیجه ، در صورت ۰ یا ۰۰ نتیجه موفق و در غیر اینصورت نا موفق می باشد
Message	شرح نتیجه
Data	اطلاعات گزارش که به شرح جدول زیر می باشد



نام پارامتر	مفهوم
MerchantID	کد پذیرنده
TerminalID	کد پایانه
InvoiceNo	شناسه پرداخت
InvoiceDate	تاریخ شمسی
Amount	مبلغ تراکنش
RefrenceNo	شماره ارجاع
ExtraInf	اطلاعات اضافی
CardNo	شماره کارت به صورت مختصر
AppExtraInf	اطلاعات اضافی مورد نیاز برنامه مانند نام پرداخت کننده، شماره همراه پرداخت کننده و ... که با فرمت JSON می باشد. <pre> { "PayerNm": "نام پرداخت کننده", "PayerMobile": "موبایل", "PayerEmail": "ایمیل", "Descr": "توضیحات", "PayerIP": "آی پی دستگاه پرداخت کننده", "PayTitle": "عنوان پرداخت" } </pre>

آماده سازی رشته SignData





راهنمای نحوه اتصال به درگاه پرداخت اینترنتی - سیز پی

برای آماده سازی این رشته لازم است نخست رشته ای ایجاد کنید از ۴ پارامتر زیر که صرفاً با یک کاراکتر کامای لاتین (,) از یکدیگر جدا شوند. توجه کنید که Delimiter تنها کاراکتر کاماست و استفاده از سایر کاراکترها همانند فاصله و غیره برای جداسازی پارامترها مجاز نیست.

این ۴ پارامتر می بایست صرفاً با یک ترتیب مشخص در رشته قرار بگیرند و پذیرنده مجاز به تغییر محل نسبی قرارگیری پارامتری در درون رشته نمی باشد (از چپ به راست).

$S=P1,P2,P3,P4$

نکته مهم: چنانچه به اقتضا شرایط بخواهید فیلدی را خالی بگذارید لازم است در هر حال و هر شرایطی کاراکتر Delimiter را درج نمائید.

روش رمز کردن رشته

روش رمز کردن به صورت زیر می باشد:

۱- ابتدا رشته بدست آمده (S) را با الگوریتم SHA256 رمز نگاری نمایید (H)

۲- سپس کد رمز نگاری شده مرحله قبل (H) را با جدا کننده کاما لاتین (,) به انتهای رشته (S)

اضافه نمایید (F) $F=S,H$

۳- در انتها عبارت بدست آمده (F) را با الگوریتم AES رمز نگاری نمایید



وب سرویس Soap (Simple Object Access Protocol)

در بخش زیر به شرح جزئیات متدهای مورد استفاده فرآیند خرید از طریق وب سرویس Soap پرداخته می شود. با بکارگیری این فناوری، امکان پیاده سازی سرویس های عملیات پرداخت برای پذیرندگان اینترنتی با سهولت زیاد فراهم می شود و ضمن آن، استقلال این فناوری از بستر پیاده سازی و استفاده از استاندارد HTTP برای تبادل ارتباطات و بهره گیری از گواهینامه امنیتی SSL معتبر، امکان مجتمع سازی امن این خدمات را روی بستر استاندارد امکان پذیر می کند. در سامانه درگاه پرداخت اینترنتی سیزپی، متدهایی در اختیار پذیرنده قرار می گیرد که در سایت خود از آنها استفاده نماید. پذیرنده قادر خواهد بود این متدها را بصورت مستقیم از داخل کد وب سایت خود فراخوانی نماید.

انتقال اطلاعات بین وب سایت پذیرنده و سرویس درگاه پرداخت اینترنتی با پروتکل Simple Object Access Protocol (SOAP) خواهد بود.

جهت استفاده از وب سرویس مزبور باید آدرس زیر در داخل کد وب سایت خود در دسترس باشد:

<https://rt.sizpay.com/KimialPGRouteService.asmx>

متد دریافت توکن (GetToken)

این متد در شروع فرآیند پرداخت جهت دریافت توکن مورد استفاده قرار می گیرد و ورودی ها و خروجی های این متد به شرح ذیل است:

ورودی متد

نام پارامتر	مفهوم	مثال
UserName	کلید رمزنگاری (KEY) (کلید ۱) از طریق پیامک برای پذیرنده ارسال شده است	
Password	IV کلید رمزنگاری (کلید ۲) از طریق پیامک برای پذیرنده ارسال شده است	
MerchantID	شماره پذیرنده	
TerminalID	شماره ترمینال	
Amount	مبلغ فاکتور به ریال بدون هیچ Separator ای می بایست ارسال شود.	۱۰۰۰



راهنمای نحوه اتصال به درگاه پرداخت اینترنتی - سیز پی

۱۳۹۵/۱۱/۲۲	تاریخ شمسی فاکتور با فرمت ۱۳۰۰/۱۲/۲۹ هر فرمتی به جز فرمت ذکر شده قابل قبول نمی باشد. صحیح بودن تاریخ بررسی می شود	DocDate
	شناسه خرید لازم است پذیرنده در هنگام ارسال درخواست خرید خود، شناسه منحصر به فرد را ارسال نماید. این شناسه باید در هر روز به صورت یکتا باشد (ترجیحا برای پیگیری های بهتر همیشه یکتا و منحصر به فرد باشد)	OrderID
	در این فیلد لازم است آدرس Callback ای که پاسخ درگاه پرداخت به آن صفحه ارسال خواهد شد را ارسال کنید. این آدرس می بایست با HTTP:// و یا HTTPS:// شروع شود (صحت آدرس چک می شود) و اکیدا توصیه می شود از شروع کردن URL ها با آدرس IP اجتناب بفرمائید.	ReturnURL
	اطلاعات اضافی، به صورت رشته می باشد (حداکثر تا ۱۰۰ کاراکتر)	ExtraInf
	شناسه پرداخت به صورت رشته ای کاملا عددی و حداکثر طول ۵۰ می باشد.	InvoiceNo
	اطلاعات اضافی مورد نیاز برنامه مانند نام پرداخت کننده، شماره همراه پرداخت کننده و ... که با فرمت JSON می باشد. { "PayerNm": "نام پرداخت کننده", "PayerMobile": "موبایل", "PayerEmail": "ایمیل", "Descr": "توضیحات", "PayerIP": "آی پی دستگاه پرداخت کننده", "PayTitle": "عنوان پرداخت" }	AppExtraInf
	لازم به ذکر است حتما باید فرمت زیر رعایت شود. در صورتی که هریک از موارد بالا مورد نیاز نمی باشد رشته تهی ارسال نمایید.	
	رشته تهی ارسال شود	SignData



خروجی متد

نام پارامتر	مفهوم
ResCod	کد نتیجه متد، در صورت ۰ یا ۰۰ نتیجه موفق و در غیر اینصورت نا موفق می باشد
Message	شرح نتیجه متد
Token	توکن. در صورت موفقیت تراکنش رشته توکن بازگشت داده می شود. توکن ارسالی یکبار مصرف بوده و به مدت محدود (حداکثر ۱۰ دقیقه) قابل استفاده می باشد

راهنمای پذیرنده (Redirect) به صفحه درگاه پرداخت اینترنتی

برای ارجاع مشتری به درگاه پرداخت، لازم است مقادیر زیر را به صورت POST به آدرس زیر ارسال نمایید با این کار مشتری پذیرنده به درگاه پرداخت وارد شده و آماده انجام تراکنش مالی می گردد.

<https://rt.sizpay.com/Route/Payment>

نام پارامتر	مفهوم
UserName	کلید رمزنگاری (KEY) (کلید ۱) از طریق پیامک برای پذیرنده ارسال شده است
Password	IV کلید رمزنگاری (کلید ۲) از طریق پیامک برای پذیرنده ارسال شده است
MerchantID	شماره پذیرنده
TerminalID	شماره ترمینال
Token	توکن
SignData	رشته تهی ارسال شود



راهنمای نحوه اتصال به درگاه پرداخت اینترنتی - سیز پی

برای راحتی کار می توانید از تابع زیر برای انجام عملیات ارجاع به درگاه پرداخت استفاده کنید. مبرهن است که این تابع را باید برای کاربرد خاص خود تغییر دهید تا متناسب با سناریوی مدنظرتان گردد:

```
<script language="javascript" type="text/javascript">
function postData() {
    var form = document.createElement("form");
    form.setAttribute("method", "POST");
    form.setAttribute("action", "https://rt.sizpay.com/Route/Payment");
    form.setAttribute("target", "_self");
    var hiddenField = document.createElement("input");
    hiddenField.setAttribute("name", "UserName");
    hiddenField.setAttribute("value", UserNameValue);
    form.appendChild(hiddenField);
    var hiddenField2 = document.createElement("input");
    hiddenField2.setAttribute("name", "Password");
    hiddenField2.setAttribute("value", PasswordValue);
    form.appendChild(hiddenField2);
    var hiddenField3 = document.createElement("input");
    hiddenField3.setAttribute("name", "MerchantID");
    hiddenField3.setAttribute("value", MrchIDValue);
    form.appendChild(hiddenField3);
    var hiddenField4 = document.createElement("input");
    hiddenField4.setAttribute("name", "TerminalID");
    hiddenField4.setAttribute("value", TrmnlIDValue);
    form.appendChild(hiddenField4);
    var hiddenField5 = document.createElement("input");
    hiddenField5.setAttribute("name", "Token");
    hiddenField5.setAttribute("value", TokenValue);
    form.appendChild(hiddenField5);
    var hiddenField6 = document.createElement("input");
    hiddenField6.setAttribute("name", "SignData");
    hiddenField6.setAttribute("value", "");
    form.appendChild(hiddenField6);
    document.body.appendChild(form);
    form.submit();
    document.body.removeChild(form);
}
</script>
```

ارسال نتیجه تراکنش به آدرس بازگشت پذیرنده

پس از تکمیل فرآیند پرداخت توسط کاربر مقادیر زیر و با متد POST به آدرس بازگشت اعلام شده توسط پذیرنده ارسال می گردد و سامانه پذیرنده می بایست در صورت موفق بودن پرداخت نسبت ارسال تاییدیه یا تراکنش بازگشت اقدام نماید.

نام پارامتر	مفهوم
ResCod	کد نتیجه پرداخت، در صورت ۰ یا ۰۰ نتیجه موفق و در غیر اینصورت نا موفق می باشد



راهنمای نحوه اتصال به درگاه پرداخت اینترنتی - سیز پی

شرح نتیجه پرداخت	Message
شماره پذیرنده	MrchID
شماره ترمینال	TrmnID
شناسه پرداخت	InvoiceNo
اطلاعات اضافی	ExtraInf
اطلاعات اضافی مورد نیاز برنامه مانند نام پرداخت کننده، شماره همراه پرداخت کننده و ... که با فرمت JSON می باشد. <pre>{ "PayerNm": "نام پرداخت کننده", "PayerMobile": "موبایل", "PayerEmail": "ایمیل", "Descr": "توضیحات", "PayerIP": "آی پی دستگاه پرداخت کننده", "PayTitle": "عنوان پرداخت" }</pre> لازم به ذکر است حتما باید فرمت زیر رعایت شود. در صورتی که هریک از موارد بالا مورد نیاز نمی باشد رشته تهی ارسال نمایید.	AppExtraInf
توکن	Token

پس از دریافت پارامترهای فوق ابتدا باید نتیجه تراکنش (ResCode) چک شود که در صورتیکه 0 یا 00 بود، کار ادامه یابد و در غیراینصورت تراکنش ناموفق بوده و وضعیت آن مختومه می باشد.

نکته مهم: در صورت عدم ارسال تاییدیه یا درخواست بازگشت در مدت تعیین شده تراکنش به صورت خودکار بازگشت داده می شود.

متد تایید پرداخت (Confirm)

پس از اطلاع یافتن پذیرنده از موفق بودن تراکنش مالی در درگاه پرداخت، لازم است پذیرنده بلافاصله این وب سرویس متد را فراخوانی نماید بدیهی است در پذیرنده فقط در صورت موفق بودن پرداخت مجاز به فراخوانی این متد می باشد.

ورودی ها و خروجی های این متد به شرح ذیل است:

ورودی متد 



راهنمای نحوه اتصال به درگاه پرداخت اینترنتی - سیز پی

نام پارامتر	مفهوم
UserName	کلید رمزنگاری (KEY) (کلید ۱) از طریق پیامک برای پذیرنده ارسال شده است
Password	IV کلید رمزنگاری (کلید ۲) از طریق پیامک برای پذیرنده ارسال شده است
MerchantID	شماره پذیرنده
TerminalID	شماره ترمینال
Token	توکن
SignData	رشته تهی ارسال شود

خروجی متد

نام پارامتر	مفهوم
ResCod	کد نتیجه پرداخت، در صورت ۰ یا ۰۰ نتیجه موفق و در غیر اینصورت نا موفق می باشد
Message	شرح نتیجه پرداخت
MrchID	شماره پذیرنده
TrmnID	شماره ترمینال
OrderID	شناسه خرید
TrnsNo	شماره تراکنش (اختصاصی)
RefNo	شماره ارجاع
TraceNo	شماره پیگیری
ExtraInf	اطلاعات اضافی
Amount	مبلغ تراکنش
CardNo	شماره کارت به صورت مختصر
TarnsDate	تاریخ و ساعت میلادی تراکنش
InvoiceNo	شناسه پرداخت
AppExtraInf	اطلاعات اضافی مورد نیاز برنامه مانند نام پرداخت کننده، شماره همراه پرداخت کننده و ... که با فرمت JSON می باشد.



راهنمای نحوه اتصال به درگاه پرداخت اینترنتی - سیز پی

<pre>{ "PayerNm": "نام پرداخت کننده", "PayerMobile": "موبایل", "PayerEmail": "ایمیل", "Descr": "توضیحات", "PayerIP": "آی پی دستگاه پرداخت کننده", "PayTitle": "عنوان پرداخت" }</pre>	
توکن	Token

موارد زیر در رابطه با فراخوانی این متد حائز اهمیت هستند:

- ❖ یک تراکنش را نمی تواند بیش از یکبار Confirm نمود.
- ❖ تراکنشی که برایش درخواست بازگشت مبلغ تراکنش شده را نمی توان Confirm نمود.
- ❖ تراکنش مالی غیرموفق را نمی توان Confirm نمود.
- ❖ عدم ارسال درخواست Confirm تا ۱۰ دقیقه بعد از انجام تراکنش (به وقت سرور پرداخت) موجب خواهد شد تا بصورت خودکار تراکنش مزبور Reverse شود و مبلغ آن به حساب مشتری برگشت داده می شود.
- ❖ همانگونه که از جدول کدهای نتیجه فراخوانی این متد پیداست، نتیجه 0 یا 00 به معنای موفقیت آمیز بودن عملیات است.

متد بازگشت پرداخت (Reverse)

چنانچه پذیرنده بخواهد مبلغ تراکنش مالی موفق را به حساب پذیرنده بازگرداند، به شرط آنکه قبلاً تراکنش مزبور را Confirm ننموده باشد، لازم است از متد Reverse استفاده کند. با فراخوانی این متد، درخواست پذیرنده برای پردازش و عودت وجه در سیکل پردازش سرور پرداخت قرار می گیرد. ورودی ها و خروجی های این متد به شرح ذیل است:

ورودی متد ➦



راهنمای نحوه اتصال به درگاه پرداخت اینترنتی - سیز پی

نام پارامتر	مفهوم
UserName	کلید رمزنگاری (KEY) (کلید ۱) از طریق پیامک برای پذیرنده ارسال شده است
Password	IV کلید رمزنگاری (کلید ۲) از طریق پیامک برای پذیرنده ارسال شده است
MerchantID	شماره پذیرنده
TerminalID	شماره ترمینال
Token	توکن
SignData	رشته تهی ارسال شود

+ خروجی متد

نام پارامتر	مفهوم
ResCod	کد نتیجه پرداخت، در صورت ۰ یا ۰۰ نتیجه موفق و در غیر اینصورت نا موفق می باشد
Message	شرح نتیجه پرداخت
MrchID	شماره پذیرنده
TrmnID	شماره ترمینال
OrderID	شناسه خرید
TrnsNo	شماره تراکنش (اختصاصی)
RefNo	شماره ارجاع
TraceNo	شماره پیگیری
ExtraInf	اطلاعات اضافی
Amount	مبلغ تراکنش
CardNo	شماره کارت به صورت مختصر
TarnsDate	تاریخ و ساعت میلادی تراکنش
InvoiceNo	شناسه پرداخت
AppExtraInf	اطلاعات اضافی مورد نیاز برنامه مانند نام پرداخت کننده، شماره همراه پرداخت کننده و ... که با فرمت JSON می باشد.



راهنمای نحوه اتصال به درگاه پرداخت اینترنتی - سیز پی

<pre>{ "PayerNm": "نام پرداخت کننده", "PayerMobile": "موبایل", "PayerEmail": "ایمیل", "Descr": "توضیحات", "PayerIP": "آی پی دستگاه پرداخت کننده", "PayTitle": "عنوان پرداخت" }</pre>	
توکن	Token

شرح سرویس های گزارش

متد گزارش اطلاعات پرداخت با شناسه پرداخت
ورودی ها و خروجی های این متد به شرح ذیل است:
ورودی متد

نام پارامتر	مفهوم
UserName	کلید رمزنگاری (KEY) (کلید ۱) از طریق پیامک برای پذیرنده ارسال شده است
Password	IV کلید رمزنگاری (کلید ۲) از طریق پیامک برای پذیرنده ارسال شده است
MerchantID	شماره پذیرنده
TerminalID	شماره ترمینال
InvoiceNo	شناسه پرداخت
InvoiceDate	تاریخ شمسی فاکتور با فرمت ۱۳۰۰/۱۲/۲۹ هر فرمتی به جز فرمت ذکر شده قابل قبول نمی باشد. صحیح بودن تاریخ بررسی می شود
SignData	رشته تهی ارسال شود

خروجی متد

نام پارامتر	مفهوم
ResCod	کد نتیجه ، در صورت ۰ یا ۰۰ نتیجه موفق و در غیر اینصورت نا موفق می باشد
Message	شرح نتیجه
Data	اطلاعات گزارش که به شرح جدول زیر می باشد

نام پارامتر	مفهوم
MerchantID	کد پذیرنده



راهنمای نحوه اتصال به درگاه پرداخت اینترنتی - سیز پی

کد پایانه	TerminalID
شناسه پرداخت	InvoiceNo
تاریخ شمسی فاکتور	InvoiceDate
مبلغ تراکنش	Amount
شماره ارجاع	RefrenceNo
اطلاعات اضافی	ExtraInf
شماره کارت به صورت مختصر	CardNo
اطلاعات اضافی مورد نیاز برنامه مانند نام پرداخت کننده، شماره همراه پرداخت کننده و ... که با فرمت JSON می باشد. <pre>{ "PayerNm": "نام پرداخت کننده", "PayerMobile": "موبایل", "PayerEmail": "ایمیل", "Descr": "توضیحات", "PayerIP": "آی پی دستگاه پرداخت کننده", "PayTitle": "عنوان پرداخت" }</pre> لازم به ذکر است حتما باید فرمت زیر رعایت شود. در صورتی که هریک از موارد بالا مورد نیاز نمی باشد رشته تهی ارسال نمایید.	AppExtraInf



اتصال به صفحه درگاه پرداخت شخصی به صورت مستقیم از App پذیرنده

برای اتصال به درگاه پرداخت شخصی، لازم است مقادیر زیر را به صورت POST به آدرس درگاه پرداخت شخصی ارسال نمایید با این کار مشتری پذیرنده به درگاه پرداخت واسط وارد شده و آماده انجام تراکنش مالی می گردد.

<https://mershant.sizpay.com/Shop/Index/ID>

نام پارامتر	مفهوم
Amount	مبلغ تراکنش
PayID	شناسه پرداخت
BackURL	آدرس بازگشت از صفحه رسید

نکات مهم:

- 1- ID در آدرس بالا همان ID موجود در لینک درگاه پرداخت شخصی می باشد.
 - 2- به یاد داشته باشید با ارسال هریک از موارد موجود در جدول بالا گزینه متناظر با آن در صفحه درگاه پرداخت شخصی به صورت فقط خواندنی تبدیل می شود و غیر قابل ویرایش می باشد.
 - 3- در صورت عدم نیاز به هریک از موارد بالا مقدار آن را با رشته تهی پر کنید
- برای راحتی کار می توانید از تابع زیر برای انجام عملیات ارجاع به درگاه پرداخت استفاده کنید. مبرهن است که این تابع را باید برای کاربرد خاص خود تغییر دهید تا متناسب با سناریوی مدنظرتان گردد:

```
<script language="javascript" type="text/javascript">
function postData() {
    var form = document.createElement("form");
    form.setAttribute("method", "POST");
    form.setAttribute("action", "https://mershant.sizpay.com/Shop/Index/ID");
    form.setAttribute("target", "_self");
    var hiddenField = document.createElement("input");
    hiddenField.setAttribute("name", "Amount");
    hiddenField.setAttribute("value", AmountValue);
    form.appendChild(hiddenField);
    var hiddenField1 = document.createElement("input");
    hiddenField1.setAttribute("name", "PayID");
    hiddenField1.setAttribute("value", PayIDValue);
    form.appendChild(hiddenField1);
    var hiddenField2 = document.createElement("input");
    hiddenField2.setAttribute("name", "BackURL");
    hiddenField2.setAttribute("value", BackURLValue);
    form.appendChild(hiddenField2);
    document.body.appendChild(form);
    form.submit();
    document.body.removeChild(form);
}
</script>
```