مستندات فنی

روش جدید اتصال به درگاه پرداخت اینترنتی

فهرست مطالب

مقدمه

خرید اینترنتی یکی از تراکنش‌های کارتی است که در مرکز شاپرک نیز جزو تراکنش‌های مجاز محسوب می‌شود. در این مستند گام‌های لازم برای ایجاد بستر پرداخت الکترونیکی در سمت وب سایت پذیرنده شرح داده شده است. این خدمت پذیرندگان را قادر می‌سازد تا از طریق درگاه پرداخت اینترنتی شرکت پرداخت الکترونیک پاسارگاد به خریداران خود سرویس ارائه دهد. این نسخه از سرویس در مقایسه با نسخه‌های پیشین تفاوت‌های بسیاری از لحاظ معماری و عملکرد را دارا می‌باشد، به گونه‌ای که تمامی متدهای ثبت درخواست‌خرید، استعلام وضعیت خرید، برگشت و تایید خرید در قالب RESTful API ارائه شده‌اند. علاوه بر این کلیه پارامترهای ورودی و خروجی وب‌سرویس از نوع JSON بوده و در زمان هدایت خریدار از وبسایت پذیرنده به درگاه پرداخت تنها می‌بایست مقدار توکن به درگاه پرداخت ارسال گردد. از این رو حجم داده‌های ارسالی به درگاه پرداخت کاهش قابل توجهی پیدا کرده‌است.

۱. تعاریف

۱-۱ تعاریف مرتبط با خریدار

- **خریدار**: هویتی است که توسط یکی از انواع کارت‌های بانکی عضو شبکه شتاب و با مراجعه به وب سایت مورد نظر خود تقاضای خرید کالا یا خدمات را دارد.

۱-۲ تعاریف مرتبط با پذیرنده

- **پذیرنده**: هویتی است که با آماده‌سازی بستر پرداخت اینترنتی، اقدام به فروش کالا و خدمات از طریق وب سایت خود می‌نماید.
- **شماره شناسائی پذیرنده (MerchantCode)**: کدی است که توسط بانک به پذیرنده اختصاص می‌یابد و در حین انجام تراکنش برای شناسایی پذیرنده از آن استفاده می‌گردد.
- **شماره شناسائی ترمینال (TerminalCode)**: کدی است که توسط بانک به پذیرنده اختصاص می‌یابد و در حین انجام تراکنش از آن استفاده می‌گردد.
- **کلید خصوصی پذیرنده (PrivateKey)**: کلیدی است که پذیرنده برای احراز هویت از آن استفاده می‌کند و تمامی داده‌های ارسالی خود به بانک را با آن کلید، امضای دیجیتال می‌کند.
- **کلید عمومی پذیرنده (PublicKey)**: کلیدی است که بانک جهت تایید امضای دیجیتال فروشگاه از آن استفاده می‌کند.
- **سپرده فروشنده**: شماره شبای پذیرنده می باشد که جهت انجام عمل تسویه حساب به شرکت پرداخت الکترونیک پاسارگاد اعلام می شود.
- **مبلغ فاکتور(Amount)**: مبلغی می‌باشد که پذیرنده می‌خواهد از خریدار دریافت نماید.
- **شماره فاکتور (InvoiceNumber)**: هر خرید از پذیرنده باید دارای شماره فاکتور خاص خود باشد.
- **تاریخ فاکتور (InvoiceDate)**: تاریخ فاکتور خرید است و فرمت آن به انتخاب فروشگاه می‌باشد (لازم به ذکر است که تاریخ و شماره فاکتور، باید به‌گونه‌ای تخصیص داده شوند که از ترکیب آنها شناسه یکتایی بدست آید تا همیشه بتوان برای شناسایی یک تراکنش خرید از آن استفاده کرد).

- **امضای دیجیتال (DigitalSignature):** امضای دیجیتال روشی مبتنی بر الگوریتم های رمزنگاری نا متقارن می‌باشد که به کمک آن می‌توان اطمینان حاصل کرد که داده‌های ارسالی از جانب فروشگاه مشخصی ارسال شده است.

- **آدرس بازگشت (RedirectAddress):** آدرس صفحه‌ای در سایت پذیرنده است که خریدار پس از انجام عملیات خرید به آن فرستاده می‌شود.

- **Timestamp:** زمان ارسال داده به درگاه پرداخت را Timestamp می‌گویند که فرمت آن به شکل "YYYY/MM/DD HH:MM:SS" بوده و به تاریخ میلادی ارسال می‌گردد. اگر هرکدام از عددهای ماه، روز، ساعت، دقیقه یا ثانیه یک رقمی باشد با قراردادن یک صفر در سمت چپ آن باید عدد دو رقمی تولید شده و برای بانک ارسال شود. نکته‌ی مهم در اینجا این است که مقداری که در فیلد Timestamp قرار می‌گیرد باید دقیقا با مقداری که تحت همین عنوان در امضای دیجیتال قرار می‌گیرد یکی باشد، همچنین هیچ دو درخواستی، نمی‌توانند دارای Timestamp یکسان باشند.

- **تلفن همراه (Mobile):** شماره تلفن همراه خریدار است که توسط فروشگاه برای درگاه پرداخت ارسال می‌گردد. در صورت ارسال این فیلد توسط فروشگاه، درگاه پرداخت، شماره کارت‌های ثبت شده کاربر در سامانه پیوند را نمایش می‌دهد و در غیراینصورت هیچ شماره کارتی نمایش داده نمی‌شود. همچنین در صورت ارسال این فیلد، کاربر می‌تواند اطلاعاتی از قبیل شماره کارت و تاریخ انقضای کارت خود را در درگاه پرداخت جهت سهولت انجام خرید در مراجعات بعدی ذخیره نماید.
نکته: قالب صحیح ارسال شماره تلفن همراه به صورت "۹۱۲۴۴۴۲۲۱۱" می‌باشد.

- **پست الکترونیک (Email):** آدرس پست الکترونیک خریدار است که می تواند توسط وب سایت فروشگاه به درگاه پرداخت ارسال شود. در صورت ارسال این فیلد توسط فروشگاه، فیلد ایمیل در درگاه پرداخت با آدرس ارسالی پر می‌شود.

- **نام پذیرنده (MerchantName):** نام پذیرنده است که می‌تواند توسط وب سایت فروشگاه به درگاه پرداخت ارسال شود. در صورت ارسال این فیلد توسط فروشگاه، فیلد نام فروشگاه در درگاه پرداخت با نام ارسالی پر می‌شود، در غیراینصورت نامی که قبلا در هنگام ثبت‌نام فروشگاه در شرکت پرداخت الکترونیک پاسارگاد ثبت شده، نمایش داده می‌شود. لازم به ذکر است در صورتی که پذیرنده می خواهد از این قابلیت استفاده نماید، می بایست درخواست فعالسازی آن را به شرکت پرداخت الکترونیک پاسارگاد بدهد.

- **شناسه پرداخت (PIDN):** شناسه خرید مورد نظر پذیرنده است که در صورت نیاز می‌بایست با فرمت عنوان شده در پیوست ۵ به درگاه پرداخت اینترنتی ارسال شود.

۳-۱ تعاریف مرتبط با درگاه پرداخت و عملیات مالی

- **درگاه پرداخت اینترنتی شرکت پرداخت الکترونیک پاسارگاد(Internet Payment Gateway):** سایتی است متعلق به شرکت پرداخت الکترونیک پاسارگاد که در آن خریدار پس از انتخاب موارد مورد خرید خود در سایت پذیرنده ، به آنجا هدایت می‌شود و در آنجا مشخصات کارت و رمز خود را وارد می‌نماید، سپس درگاه پرداخت تراکنش مورد نظر خریدار را انجام داده و در نهایت پذیرنده را از نتیجه آن آگاه می‌سازد.

نکته: شرکت پرداخت الکترونیک پاسارگاد ازین پس با نام پپ (Pep) در این مستند ذکر خواهد شد.

- **نوع تراکنش (Action):** نشان دهنده‌ی نوع عملیات مالی مورد نظر می‌باشد که در این سیستم کد ۱۰۰۳ جهت عملیات خرید می بایست ارسال گردد.
- **شماره رهگیری (TransactionReferenceID):** شماره‌ای است که درگاه پپ پس از موفقیت‌آمیز بودن تراکنش به سایت پذیرنده ارسال می‌کند و به وسیله آن پذیرنده می‌تواند از موفقیت آمیز بودن تراکنش اطلاع یابد.

- **تسویه حساب**: واریز وجوه دریافتی از خریدار به سپرده پذیرنده توسط پپ می‌باشد که در صورت موفق بودن تراکنش خرید پس از کسر کارمزد انجام می‌شود.

- **Token:** مقداری است که در پاسخ فراخوانی سرویس GetToken به پذیرنده برگردانده می‌شود و جهت آغاز فرایند خرید پذیرنده می‌بایست خریدار خود را به همراه توکن دریافتی به درگاه پپ پرداخت هدایت نماید.

۲. مراحل کلی فرآیند خرید

۲-۱ ورود مشتری به سایت پذیرنده و تشکیل سبد خرید یا انتخاب خدمات مورد نظر.

۲-۲ فراخوانی متد GetToken و دریافت توکن توسط پذیرنده.

۲-۳ هدایت مشتری به درگاه پپ به همراه توکن دریافتی.

۲-۴ تکمیل اطلاعات کارت توسط مشتری و انجام مرحله‌ی پرداخت وجه.

۲-۵ هدایت مشتری به سایت پذیرنده پس از انجام عملیات پرداخت توسط مشتری.

۲-۶ ادامه روند عملیات خرید مشتری در سایت پذیرنده پس از دریافت نتیجه تراکنش از طریق وب‌سرویس.

۲-۷ نهایی کردن تراکنش خرید پس از ارائه‌ی خدمات به خریدار. پذیرنده می‌بایست با استفاده از وب‌سرویس تایید خرید تراکنش مربوطه را نهایی کند. لازم به ذکر است در صورت عدم تایید، مبلغ تراکنش پس از مهلت تعیین شده به حساب مشتری عودت پیدا خواهد کرد.

۲-۸ در هر صورت، اگر پذیرنده قصد ابطال تراکنش را داشته باشد می‌تواند تا ۲ساعت بعد از تراکنش به وسیله وب‌سرویس ارائه شده اقدام به برگشت خرید نماید.

۳. دریافت توکن پرداخت

خریدار با مراجعه به وب‌سایت پذیرنده و انتخاب کالا یا خدمات مورد نیاز، آماده پرداخت مبلغ فاکتور می‌شود. سایت پذیرنده می‌بایست اطلاعات مربوط به تراکنش خرید را با PrivateKey خود امضا کرده و از طریق آدرس https://pep.shaparak.ir/Api/v1/Payment/GetToken به وب‌سرویس پپ ارسال نماید. درصورتی‌که اطلاعات ارسالی از جانب پذیرنده معتبر باشد توکن تولید شده به پذیرنده برگردانده خواهد شد. پذیرنده در مرحله بعد می‌تواند با توکن دریافتی، مشتری خود را به درگاه پپ هدایت نماید. جزئیات فراخوانی این متد در جدول زیر شرح داده شده است.

آدرس	https://pep.shaparak.ir/Api/v1/Payment/GetToken		
متد	POST		
	نام فیلد (موارد ستاره‌دار ضروری می‌باشند)	نوع فیلد	توضیحات
	Content-Type *	String	فرمت محتوی درخواست Application/json
	Sign *	String	امضای دیجیتال پارامترهای ورودی(پیوست ۲)
	InvoiceNumber *	String	شماره فاکتور
	InvoiceDate *	String	تاریخ فاکتور
	TerminalCode *	Int	کد ترمینال
	MerchantCode *	Int	کد پذیرنده
	Amount *	Decimal	مبلغ تراکنش
	RedirectAddress *	String	آدرس بازگشت به سایت پذیرنده
	Timestamp *	String	زمان ارسال تراکنش با فرمت 2018/09/18 15:15:13
	Action *	Int	خرید ۱۰۰۳
	Mobile	String	شماره موبایل خریدار
	Email	String	ایمیل خریدار
	MerchantName	String	در صورت داشتن مجوز ارسال شود
	PIDN	String	شناسه پرداخت(پیوست ۵)
	IsSuccess	Bool	False | True
	Message	String	پیغام
	Token	String	توکن تولید شده جهت عملیات پرداخت

نمونه Json	ورودی	خروجی

		موفق
		{
		"IsSuccess": true,
		"Message": "عملیات با موفقیت انجام شد",
		"Token": "02302dasd15a1f121fasd2asda"
		}
		ناموفق
		{
		"IsSuccess": false,
		"Message": "تراکنش ارسالي معتبر نیست"
		}

کدپاسخ HTTP		موفق: **200 Ok**

۴. ارجاع خریدار به درگاه پرداخت پپ

در صورتی‌که دریافت توکن با موفقیت انجام شد، سایت پذیرنده می‌بایست خریدار را همراه با توکن دریافتی به درگاه پرداخت هدایت کند. پذیرنده می‌تواند با یکی از دو روش زیر خریدار را به درگاه پرداخت هدایت نماید.

۴-۱ ارسال خریدار به درگاه با استفاده از متد Get

پس از دریافت توکن، پذیرنده می‌تواند خریدار را با آدرس https://pep.shaparak.ir/payment.aspx?n=Token به درگاه پرداخت پپ هدایت نماید. لازم به ذکر است که در آدرس ذکر شده به‌جای Token می‌بایست مقدار توکن دریافتی (در مرحله دریافت توکن پرداخت) جایگذاری گردد.

۴-۲ ارسال خریدار به درگاه با استفاده از متد Post

در صورتی که دریافت توکن با موفقیت انجام شد، سایت پذیرنده می‌بایست توکن دریافتی را با متد HTTP POST به سایت پپ https://pep.shaparak.ir/payment.aspx ارسال نماید. فرم ارسالی از سایت پذیرنده می‌بایست شامل فیلدی با مقدار توکن دریافتی و با نام Token باشد. نمونه کد HTML مورد نیاز در پیوست ۴ آورده شده است.
مشتری پس از هدایت شدن به درگاه پپ با وارد کردن شماره کارت (PAN)، کلمه عبور اینترنتی (PIN2)، کد اعتبارسنجی دوم (CVV2) و تاریخ انقضای کارت (Expiration Date) اقدام به پرداخت مبلغ مورد نظر پذیرنده می‌نماید.
در این مرحله تراکنش توسط درگاه پپ پردازش گردیده و عملیات لازم در مرکز شاپرک و بانک صادرکننده کارت انجام می‌پذیرد.
وب سایت پپ پس از انجام تراکنش، مشتری را به آدرسی که در هنگام دریافت توکن پذیرنده در فیلد redirectAddress قرار داده است هدایت می‌نماید و در QueryString آن مقادیر زیر را قرار می‌دهد.

- InvoiceNumber (در فیلد iN)
- InvoiceDate (در فیلد iD)
- TransactionReferenceID (در فیلد tref)

۵. استعلام نتیجه تراکنش

سایت پذیرنده در این گام می‌تواند با فراخوانی متد CheckTransactionResult در وب‌سرویس پپ از نتیجه عملیات خرید باخبر شود. متد CheckTransactionResult به دو روش زیر قابل فراخوانی می‌باشد.

۵-۱ درصورت دریافت نتیجه تراکنش

درصورتی‌که سایت پذیرنده مقادیر ارسالی از درگاه پرداخت الکترونیک پاسارگاد به سایت پذیرنده را دریافت نموده باشد می‌تواند تنها با تکمیل فیلد TransactionReferenceID ، نتیجه‌ی تراکنش خریدار خود را دریافت نماید. جزئیات فراخوانی این متد در جدول زیر شرح داده شده است.

۵-۲ درصورت عدم دریافت نتیجه تراکنش

درصورتی‌که سایت پذیرنده به هر دلیلی در مرحله‌ی انتقال از درگاه به سایت پذیرنده موفق به دریافت TransactionReferenceID نشود می‌تواند با مقداردهی فیلدهای شماره فاکتور، تاریخ فاکتور، شماره شناسایی پذیرنده و شماره شناسایی ترمینال از نتیجه تراکنش خریدار خود باخبر شود.
سایت پپ پس از تطبیق دادن پارامترهای ارسالی با فاکتور اصلی، نتیجه تراکنش را خوانده و اقدام مقتضی را انجام می‌دهد. جزئیات فراخوانی این متد در جدول زیر شرح داده شده است.

نکته: لازم به ذکر است که پذیرنده می‌بایست نتیجه تراکنش را چک کرده و از موفق بودن تراکنش اطمینان حاصل کند و به صرف دریافت TransactionReferenceID از درگاه، تراکنش را موفقیت آمیز تلقی نکند.

آدرس	https://pep.shaparak.ir/Api/v1/Payment/CheckTransactionResult		
متد	POST		
	نام فیلد	نوع فیلد	توضیحات
	Content-Type	String	فرمت محتوی درخواست Application/json
	InvoiceNumber	String	شماره فاکتور (در حالت ۵-۲ تکمیل گردد)
	InvoiceDate	String	تاریخ فاکتور (در حالت ۵-۲ تکمیل گردد)
	TerminalCode	Int	کد ترمینال (در حالت ۵-۲ تکمیل گردد)
	MerchantCode	Int	کد مشتری (در حالت ۵-۲ تکمیل گردد)
	TransactionReferenceID	String	شماره ارجاع داخلی موجود در QueryString در زمان هدایت خریدار از صفحه درگاه پرداخت به RedirectAddress اعلام شده توسط پذیرنده پس از انجام عملیات پرداخت. (در حالت ۵-۱ تکمیل گردد)
	IsSuccess	Bool	False | True
	Message	String	پیغام
	ReferenceNumber	long	شماره ارجاع
	TraceNumber	int	شماره پیگیری
	TransactionDate	String	تاریخ تراکنش
	Action	String	خرید ۱۰۰۳
	TransactionReferenceID	String	شماره ارجاع داخلی
	InvoiceNumber	String	شماره فاکتور
	InvoiceDate	String	تاریخ فاکتور
	MerchantCode	Int	کد پذیرنده
	TerminalCode	Int	کد ترمینال
	Amount	decimal	مبلغ تراکنش
	ورودی		خروجی
			موفق

کدپاسخ HTTP	

	{
	"InvoiceNumber": "123456",
	"InvoiceDate": "1349/04/04",
	"TerminalCode": "1",
	"MerchantCode": "1"
	}
	یا
	{
	" TransactionReferenceID ": "324324234"
	}

موفق: **200 Ok**	{
	"TraceNumber": 13,
	"ReferenceNumber": 100200300400500,
	"TransactionDate": "2019/01/29 18:07:29",
	"Action": "1003",
	"TransactionReferenceID": "636843820118990203",
	"InvoiceNumber": "80",
	"InvoiceDate": "2019/01/29 18:06:43",
	"MerchantCode": 550166,
	"TerminalCode": 550363,
	"Amount": 10000,
	"IsSuccess": true,
	"Message": "عملیات به اتمام رسید"
	}

	ناموفق

	{
	"IsSuccess": false,
	"Message": "تراکنش ارسالي معتبر نيست"
	}

۶. تایید خرید

در صورتی‌که عملیات خرید با موفقیت انجام شده باشد، درگاه پپ مدت زمان مشخصی منتظر می‌ماند تا خرید انجام شده توسط فروشنده تایید شود. درصورتی‌که خرید طی این مدت زمان توسط پذیرنده تایید نشود درگاه پپ به صورت خودکار آن را برگشت زده و پول به حساب خریدار باز می‌گردد. لازم به ذکر است که مدت زمانی‌که بانک برای تایید خرید از سمت پذیرنده منتظر می‌ماند طبق توافق پپ و پذیرنده می‌باشد.

در نتیجه پذیرنده پس از دریافت نتیجه تراکنش می‌بایست از طریق فراخوانی متد VerifyPayment در وب‌سرویس پپ اقدام به تایید خرید خریدار خود نماید. توجه شود که برای تایید خرید، پذیرنده باید شماره

فاکتورو تاریخ فاکتور تراکنش خرید (تراکنش اصلی) را ارسال کند ولی TimeStamp باید با تاریخ جاری سیستم مقداردهی شود. جزئیات فراخوانی این متد در جدول زیر شرح داده شده است.

آدرس	https://pep.shaparak.ir/Api/v1/Payment/VerifyPayment		
متد	POST		
	نام فیلد	نوع فیلد	توضیحات
	Content-Type *	String	فرمت محتوی درخواست Application/json
	Sign *	String	امضای دیجیتال پارامترهای ورودی(پیوست ۲)
	InvoiceNumber *	String	شماره فاکتور
	InvoiceDate *	String	تاریخ فاکتور
	TerminalCode *	Int	کد ترمینال
	MerchantCode *	Int	کد پذیرنده
	Amount *	Decimal	مبلغ تراکنش
	TimeStamp *	String	زمان ارسال تراکنش با فرمت 2018/09/18 15:15:13
	IsSuccess	Bool	False | True
	Message	String	پیغام
	MaskedCardNumber	String	شماره کارت خریدار به صورت Mask شده
	HashedCardNumber	String	شماره کارت خریدار به صورت Hash شده
	ShaparakRefNumber	String	شماره ارجاع شاپرکی
	ورودی	خروجی	
		موفق	

```
{
  "IsSuccess": true,
  "Message": "عملیات با موفقیت انجام شد",
  "MaskedCardNumber": "5022-29**-****-2328",
  "HashedCardNumber":
        "2DDB1E270C598677AE328AA37C2970E3075E1DB6665C5AAFD13
        1C59F7FAD99F23680536B07C140D24AAD8355EA9725A5493AC48
        E0F48E39D50B54DB906958182",
  "ShaparakRefNumber": "100200300400500"
}
```

	ناموفق

```
{
  "IsSuccess": false,
  "Message": "تراکنش ارسالي معتبر نيست"
}
```

کدپاسخ HTTP	موفق: **200 Ok**

۷. برگشت از خرید

در صورتی‌که فروشنده به هر دلیلی مایل به برگشت زدن خرید باشد، می‌تواند حداکثر تا دو ساعت پس از انجام تراکنش خرید این‌کار را توسط فراخوانی متد Refund انجام دهد.

توجه شود که برای تراکنش‌های برگشت از خرید، پذیرنده باید شماره فاکتور و تاریخ فاکتور تراکنش خرید (تراکنش اصلی) را ارسال کند ولی TimeStamp باید با تاریخ جاری سیستم مقداردهی شود. جزئیات فراخوانی این متد در جدول زیر شرح داده شده است.

آدرس	https://pep.shaparak.ir/Api/v1/Payment/RefundPayment		
متد	POST		
	نام فیلد	نوع فیلد	توضیحات
	Content-Type *	String	فرمت محتوی درخواست Application/json
	Sign *	String	امضای دیجیتال پارامترهای ورودی(پیوست ۲)
	InvoiceNumber *	String	شماره فاکتور
	InvoiceDate *	String	تاریخ فاکتور
	TerminalCode *	Int	کد ترمینال
	MerchantCode *	Int	کد پذیرنده
	TimeStamp *	String	
	IsSuccess	Bool	False | True
	Message	String	پیغام
	ورودی		خروجی
			موفق

			<pre>{
			"IsSuccess": true,
			"Message": "عملیات با موفقیت انجام شد"
			}</pre>

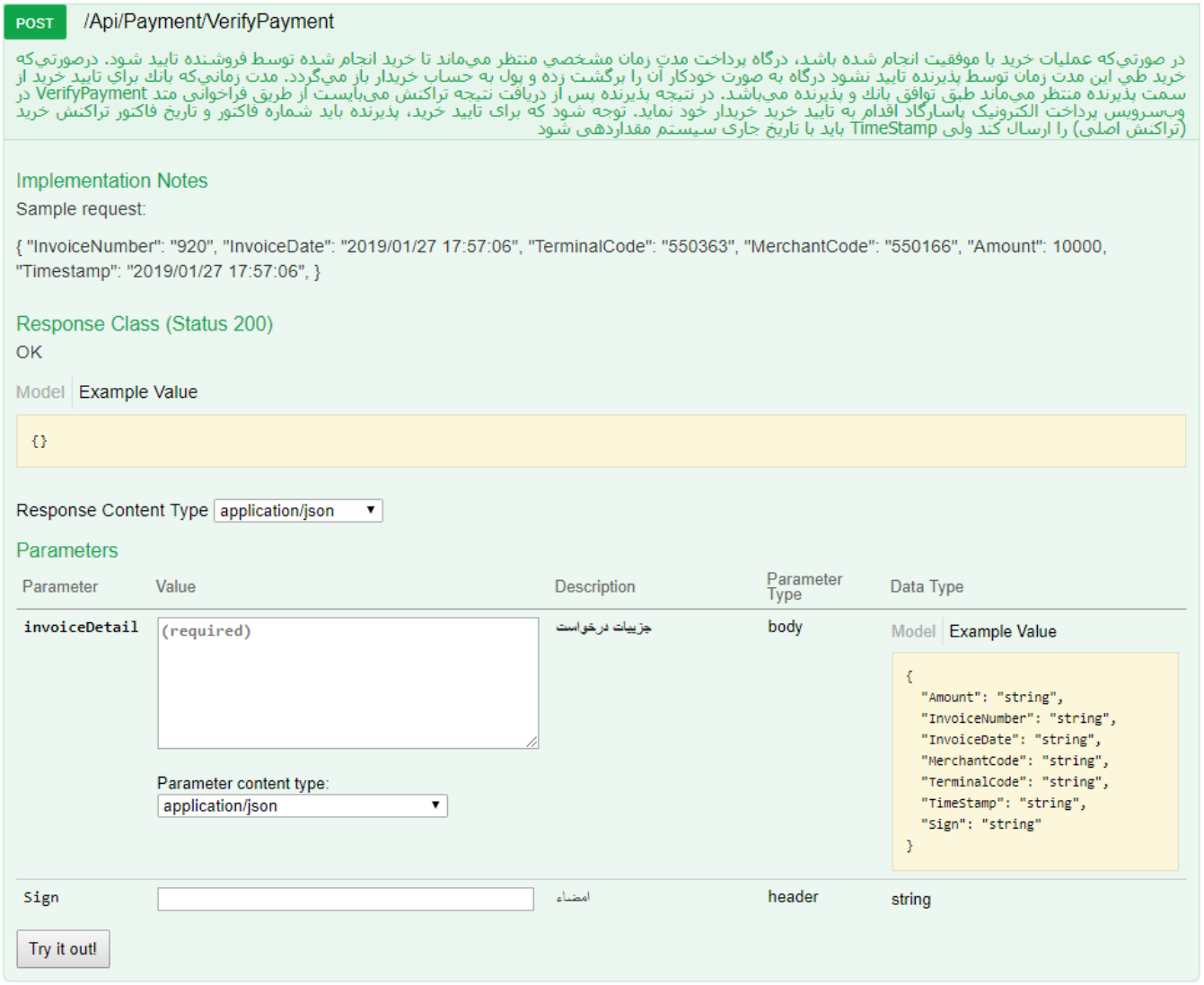
			ناموفق

			<pre>{
			"IsSuccess": false,
			"Message": "تراکنش ارسالی معتبر نیست"
			}</pre>

کد پاسخ HTTP			موفق: **200 Ok**

۸. استفاده از swagger جهت تست و فراخوانی سرویس‌ها

با استفاده از آدرس https://pep.shapark.ir/swagger/ui/index می‌توان لیست کلیه سرویس‌های پرداخت، به همراه توضیحات مربوطه، فرمت پارامترهای ورودی و خروجی و همچنین نمونه اطلاعات جهت فراخوانی وب سرویس‌ها را مشاهده کرد. به طور مثال جهت آگاهی از نحوه فراخوانی متد تایید خرید، با مراجعه به آدرس مذکور می‌توان این مورد را از لیست سرویس‌ها پیدا کرده و ورودی و خروجی‌ها را مشاهده نمود.

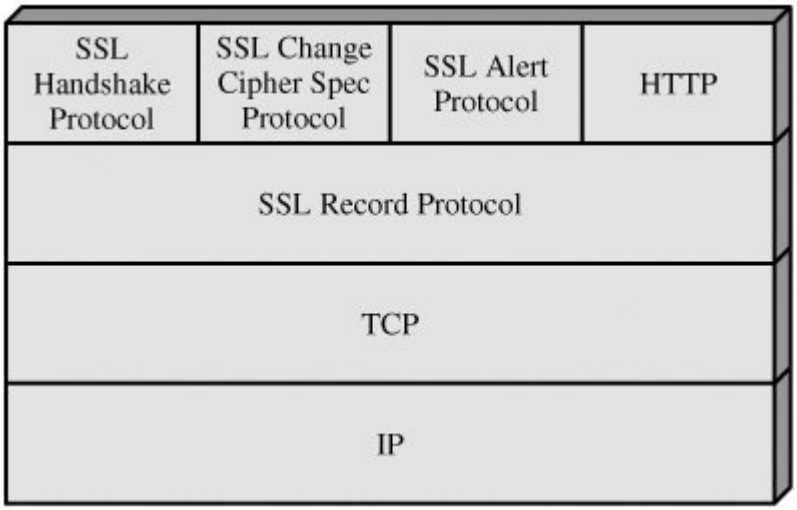


با وارد کردن پارامتر invoiceDetail در باکس مربوطه همانند نمونه ورودی(sample request) و همین‌طور پارامتر امضا و زدن دکمه Try it Out می‌توان این سرویس را فراخوانی و نتیجه آن را در همین قسمت ملاحظه نمود.

پیوست۱: نیازمندی‌های امنیتی

جهت برقراری ارتباط امن فیمابین سایت پذیرنده و سایت بانک، سایت بانک از پروتکل SSL استفاده می‌کند. پروتکل SSL (Secure Socket Layer) یک استاندارد وب برای رمزنگاری اطلاعات بین کاربر و وب سایت است. اطلاعاتی که توسط یک اتصال SSL مبادله می‌شوند بصورت رمز شده ارسال می‌شوند و بدین ترتیب اطلاعات مبادله شده از دزدیده شدن یا استراق سمع محافظت می‌شوند. SSL برای شرکت‌ها و مشتریان این امکان را فراهم می‌کند که بتوانند با اطمینان اطلاعات خود (مانند شماره کارت اعتباری و ...) را به یک وب سایت بطور محرمانه ارسال کنند. برای برقراری یک اتصال SSL نیاز به یک SSL Certificate می‌باشد.

همچنین پیشنهاد می‌شود که سایت پذیرنده نیز از پروتکل SSL استفاده کند اما اجباری نیست.



یکی دیگر از نیازمندی‌های امنیتی این است که پذیرنده نباید از هیچ‌کدام از اطلاعات مالی خریدار (همانند مشخصات کارت، کلمه رمز کارت و ...) مطلع شود. به همین دلیل پذیرنده از مشتری هیچ نوع اطلاعات مالی و بانکی دریافت نمی‌کند و تمامی این اطلاعات توسط خریدار صرفاً در درگاه پپ وارد می‌شود.

پیوست۲: نحوه ایجاد امضای دیجیتال

یکی دیگر از نیازمندی‌های امنیتی این است که پذیرنده می‌بایست تمامی بدنه درخواست که به فرمت json می‌باشد را امضا کرده و نتیجه‌ی حاصله را در یک هدر با عنوان Sign برای متد مورد نظر در وب‌سرویس ارسال نماید. با کمک این امضای دیجیتال درگاه پرداخت پپ از بابت صحت اطلاعات ارسالی از طرف پذیرنده، اطمینان حاصل می‌نماید. مثالی از نحوه‌ی ایجاد امضای دیجیتال به زبان c# در زیر آمده است.

```
public string GetSign(string data)
{
        var cs = new CspParameters { KeyContainerName = "PaymentTest" };
        var rsa = new RSACryptoServiceProvider(cs) { PersistKeyInCsp = false };

        rsa.Clear();
        rsa = new RSACryptoServiceProvider();
        rsa.FromXmlString("<RSAKeyValue><Modu-
        lus>2vW5l4pF6e2xBe4L0pq197o2/Xm6A4ktgYme+S/7SAQglFYb9swyrXzeeZGM0FDzi-
        UuxPJpCmG85As35L3PBHc5D/cVWt7HyrgoygXguzIVFLQPcS-
        kGlPhiu/9wXFl+wt579JloapjYIM99vzwsw1X8W5HMWO02snNR4Gh+KujE=</Modulus><Expo-
        nent>AQAB</Exponent><P>7TpcqQxQSnxQuGXg1gAExDjkLZRu-
        WXMsMZ1HcGs9B659ZCtyx0K9PZeBb+rSbsDiQ53ME7gfr1qez5OJs8V3Kw==</P><Q>7ElMHADjw9SjcIo
        A9ZiEtdZN-
        lesfuxImTJ3ZSgVJLHm1DzpddZizOiPMcoJB30p9DUybV0yA92Ks9vPON1emEw==</Q><DP>M5IXaKyDm9
        2wkpWbLgps/tc7S2lUH9/4wIRnblnxt4TIP4lud0Db8NLJ0bGjs239AiQAp-
        FzHjpBNq+Rv8APCZw==</DP><DQ>t1v15PFHzorvPgDJ18xAh/9Ce+lW2UuvTtt-
        baFaLvyi/OY74Qa56vs5S9DVius1KF4zwKyGiHteDE6yExWuN4w==</DQ><InverseQ>ps0bU-
        jxFx2RxXgvA8nU+C+RSGVI3UgoUMD9L+asg2YrCSj11S1RQqu4Yh+bkIY3sb58/5VpzRZ1/DufdZ1/LFw=
        =</InverseQ><D>Ohw2yioPTfsgHRPFHXqjyoAoMfNJU8DnS9arUzTRupYl01hbCY+9718Ra-
        DAhr/Ob/pcRDaVfATebaMwjUt2kpXe6R5TQ7dfShxNqbBgQtPBXvjuhoOtU6mFOWSyE-
        GQUEsVsVSexvJyM/I1aXj9m9BHVSp0gp6+Jb5yYM/G2PnbU=</D></RSAKeyValue>");

        byte[] signMain = rsa.SignData(Encoding.UTF8.GetBytes(data), new
        SHA1CryptoServiceProvider());
        string sign = Convert.ToBase64String(signMain);
        return sign;

}
```

پیوست۳: نمونه کد فراخوانی سرویس‌های درگاه پرداخت

نمونه کد فراخوانی سرویس دریافت مجوز پرداخت(Token) به زبان C# به شرح زیر می‌باشد. لازم به ذکر است جهت فراخوانی تمامی سرویس‌های درگاه پرداخت می‌توان از قطعه کد زیر استفاده نمود، به‌صورتی که تنها می‌بایست آدرس متد مورد نظر و مقدار json ورودی آن متد در کد زیر قرار گیرد.

```
public static string CallPaymentMethod()
{
    var sendingData = "{ \"InvoiceNumber\": \"123456\", \"InvoiceDate\":
    \"1349/04/04\",\"TerminalCode\": \"1\", \"MerchantCode\": \"1\", \"Amount\":\"1000\",
    \"RedirectAddress\":\" https://www.sample.com/PaymentResult\",
    \"Timestamp\":\"2018/09/18 15:15:13\", \"Action\":\"1003\",
    \"Mobile\":\"09122222222\", \"Email\":\"BuyerName@Sample.ir\" }";

    var content = new StringContent(sendingData, Encoding.UTF8, "application/json");
    var request = new HttpRequestMessage
                {
                    RequestUri = new Uri("https://pep.shaparak.ir/Api/v1/Payment/GetToken"),
                    Method = HttpMethod.Post,
                    Content = content
                };

    request.Headers.Add("Sign", GetSign(sendingData));

    var client = new HttpClient();
    client.DefaultRequestHeaders.Accept.Add(new
       MediaTypeWithQualityHeaderValue("application/json"));

    var response = client.SendAsync(requestMessage).Result;
    return Encoding.UTF8.GetString(response.Content.ReadAsByteArrayAsync().Result);
}
```

پیوست۴: نمونه کد HTML هدایت مشتری به درگاه پرداخت

سایت پذیرنده پس از دریافت توکن پرداخت از طریق متد GetToken اقدام به نمایش فرم زیر به مشتری خود می‌نماید. مشتری با کلیک بر روی گزینه "ارسال به درگاه" به درگاه پرداخت بانک منتقل شده و عملیات پرداخت خود را تکمیل می‌کند.

```
<form id="PaymentForm" method="post" Action="https://pep.shaparak.ir/payment.aspx" >
 <input type="hidden" name="Token" value="توکن دریافتی از طریق وب سرویس" />
 <input type="submit" name="submit" value="ارسال به درگاه" />
</form>
```

پیوست۵: الگوریتم اعتبار سنجی شناسه پرداخت

شناسه پرداختی که به سمت درگاه پرداخت اینترنتی ارسال می شود باید از ساختار زیر تبعیت کند تا معتبر شناخته شود، درغیر این صورت پذیرنده با پیغام خطا مواجه می شود.

- شماره شناسه: رقم کنترلی(۲ کاراکتر) + شماره سریال (بین ۳ تا ۱۷ کاراکتر) "–" شماره حساب(حداکثر ۱۰ کاراکتر)
- شماره حساب: شماره سپرده مشتری بدون نقطه
- شماره سریال: ترکیب پارامترهای تعریف شده توسط کاربر به صورت رشته ای
- رقم کنترلی: محاسبه شده بر اساس شماره حساب، سریال و مبلغ

نکته: مبلغ می بایست کمتر از ۱۵ رقم باشد.

نحوه محاسبه رقم کنترلی:

الف: کلیه کاراکتر های سریال از سمت راست در اعداد اول به شرح زیر ضرب شده و حاصل جمع آنها برابر مقدار پارامتر A خواهد بود

موقعیت مکانی از سمت راست	۱	۲	۳	۴	۵	۶	۷	۸	۹	۱۰	۱۱	۱۲	۱۳	۱۴	۱۵
اعداد اول	۵۳	۴۷	۴۳	۴۱	۳۷	۳۱	۲۹	۲۳	۱۹	۱۷	۱۳	۱۱	۷	۵	۳

ب: اعداد شماره حساب از سمت راست در اعداد اول به شرح زیر ضرب شده و حاصل جمع آنها برابر مقدار پارامتر B خواهد بود

موقعیت مکانی از سمت راست	۱	۲	۳	۴	۵	۶	۷	۸	۹	۱۰
اعداد اول	۳۷	۳۱	۲۹	۲۳	۱۹	۱۷	۱۳	۱۱	۷	۵

ج: مبلغ از سمت راست در اعداد اول به شرح زیر ضرب شده و حاصل جمع آنها برابر مقدار پارامتر C خواهد بود

موقعیت مکانی از سمت راست	۱	۲	۳	۴	۵	۶	۷	۸	۹	۱۰	۱۱	۱۲	۱۳	۱۴	۱۵
اعداد اول	۶۱	۵۹	۵۳	۴۷	۴۳	۴۱	۳۷	۳۱	۲۹	۲۳	۱۹	۱۷	۱۳	۱۱	۷

د: [(A+B+C)Mod ۹۹]

مثال:

شناسه: ۱۸۰۰۰۷۶۲۱-۱۲۳۴۲۵

سریال:۱۲۳۴

شماره حساب: ۱۸۰۰۰۷۶۲۱

مبلغ: ۱۲۵۰۰

رقم کنترلی: ۲۵

الف:

شناسه	۴	۳	۲	۱
اعداد اول	۵۳	۴۷	۴۳	۴۱
حاصلضرب	۲۱۲	۱۴۱	۸۶	۴۱
حاصلجمع	۴۸۰			

ب:

شماره حساب	۱	۲	۶	۷	۰	۰	۸	۱
اعداد اول	۳۷	۳۱	۲۹	۲۳	۱۹	۱۷	۱۳	۱۱	۷
حاصلضرب	۳۷	۶۲	۱۷۴	۱۶۱	۰	۰	۰	۸۸	۷
حاصلجمع	۵۲۲								

ج:

مبلغ	۰	۰	۵	۲	۱
اعداد اول	۶۱	۵۹	۵۳	۴۷	۴۳
حاصلضرب	۰	۰	۲۶۵	۹۴	۴۳
حاصلجمع	۴۰۲				

د: ۲۵ = [(۴۸۰+۵۲۲+۴۰۲)Mod ۹۹]

پیوست۶: الگوریتم رمز نگاری نامتقارن

الگوریتم‌های رمز گذاری نامتقارن نوعی از الگوریتم‌های رمز نگاری هستند که دارای دو کلید مختلف می‌باشند که از یکی جهت رمزنگاری و از دیگری جهت رمز گشایی استفاده می‌شود. این الگوریتم‌ها در گستره وسیعی از کاربردها به کار می رود. در این الگوریتم‌ها کلید اول را کلید عمومی (Public Key) و کلید دوم را کلید خصوصی(Private Key) می‌نامند. یکی از کاربردهای مهم الگوریتم های رمز نگاری نامتقارن استفاده از آنها در تولید امضای دیجیتال می‌باشد.

مفهوم امضای دیجیتال

امضای دیجیتال روشی مبتنی بر الگوریتم های رمزنگاری نا متقارن می باشد که به کمک ان می‌توان اطمینان حاصل کرد که داده‌های ارسالی از جانب شخص مشخصی ارسال شده است. نمونه ای از این الگوریتم‌ها می توان به RSA و DSA اشاره کرد.

روال کار در امضای دیجیتال به این شکل است که پیش از ارسال داده ها، اطلاعات را با استفاده از الگوریتم‌های درهم سازی یک‌طرفه (Hash Algorithms) به یک کد درهم (Hash) تبدیل می‌شود. از نمونه این الگوریتم ها می توان به MD5, SHA1 و ... اشاره کرد. یک‌طرفه بودن در این الگوریتم‌ها به این معنی است که پس از کد شدن اطلاعات به هیچ عنوان نمی توان از روی این کدها، اطلاعات اصلی را به دست آورد. پس از در هم سازی اطلاعات، به منظور تولید امضای دیجیتال، باید از یکی از الگوریتم‌های رمز نگاری نامتقارن استفاده شود، و با استفاده از کلید خصوصی (Private Key) آن الگوریتم، رشته‌ی تولید شده توسط الگوریتم درهم سازی را امضا نمود.

مفهوم کلید عمومی و کلید خصوصی

کلید عمومی بخشی از کلید است که بین همه توزیع می‌شود و هیچ نگرانی از لو رفتن و دزدیده شدن آن وجود ندارد به واقع لفظ "عمومی" نیز بیان‌گر همین مطلب است. اگر داده‌ای برای صاحب کلید عمومی (پخش کننده کلید عمومی) باید رمز شود با استفاده از این کلید رمز نگاری شده و ارسال می‌شود. نکته مهم الگوریتم‌های نامتقارن در این مطلب است که داده‌های رمز شده با کلید عمومی فقط و فقط با کلید خصوصی قابل رمز گشایی هستند و دوباره با همان کلید عمومی نمی‌توان آنها را رمزگشایی کرد به همین دلیل داشتن کلید عمومی کمکی به رمزگشایی داده‌ها نخواهد کرد.

کلید خصوصی در واقع بخشی از کلید است که به وسیله آن داده های رمز شده به وسیله کلید عمومی را می‌توان رمز گشایی کرد. صاحب کلید خصوصی باید حداکثر محافظت از این کلید را انجام دهد و به هیچ عنوان اجازه ندهد که این کلید در دست کسی غیر از خودش قرار گیرد. علاوه بر این با استفاده از کلید خصوصی می‌توان اسناد و مدارک مانند Documentها، Emailها و پیغامها را امضا کرد و امضای صورت گرفته را در انتهای Document، Email و یا پیغام قرار داد. در این حالت گیرنده پیغام با داشتن اصل پیغام، امضای دیجیتال زیر آن و کلید عمومی شما می تواند از صحت امضا اطمینان حاصل کند و مطمئن شود که داده‌ها از جانب شما ارسال شده است. اما با کلید عمومی به هیچ عنوان نمی‌تواند امضای شما را جعل کند.