



PTT Akıllı Esnaf

API Entegrasyon Dokümanı

İçindekiler

İçindekiler.....	2
Güncelleme Tarihçesi.....	2
1 Genel Bakış.....	3
2 Süreç	3
3 3D ile Ödeme	4
4 Ödeme Sorgulama	5
5 İptal	7
6 İade	8
7 İşlem Listeleme.....	9
8 Ortak Ödeme Sayfası	10
9 ProcessCardForm.....	11
10 Hash Mekanizması	11
11 Ortam ve Test Kart Bilgileri	11
12 Hata Kodları ve Açıklamaları.....	13
12 Sıkça Sorulan Sorular	14

Güncelleme Tarihçesi

Tarih	Açıklama	Güncelleyen	Sürüm
29.12.2020	İlk taslak	DgpaysIT	1.0
09.04.2021	ProcessCard Form methodu eklendi. Ortam bilgilerinde güncelleme yapıldı.	DgpaysIT	1.1
16.04.2021	Ortam bilgisi URL güncelleme yapıldı.	DgpaysIT	1.2
07.05.2021	Ortam biglilerine ekleme yapıldı	DgpaysIT	1.3

1 Genel Bakış

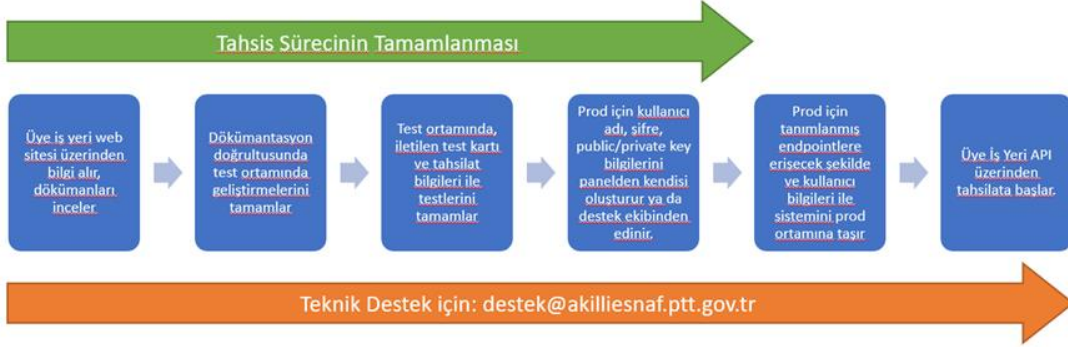
API entegrasyonunda üye işyeri kendi ödeme formunu kullanır. Bu form üzerindeki kontrolleri ve kuralları uygulaması gerekir.

- Kart numarası(PAN) luhn algoritmasına tabi tutulmalıdır.
- Kartın hamilinin adı ve soyadı alınmalıdır.
- Son kullanma tarihi geçmiş kartlar ile API'ye sorgu yapılmamalıdır.
- AMEX kartları ile işlem alabilir şekilde tasarlanmalıdır.

Entegrasyon kapsamında sorun yaşamanız durumunda destek@akilliesnaf.gov.tr mail adresinden destek alabilirsiniz.

2 Süreç

Aşağıdaki adımlarınız tamamlandıktan sonra üretim ortamında işlem yapabilirsiniz.



Tek çekim, taksitli ve tek tıkla ödeme işlemleri için uygundur. İşleme gönderilen kartın bakiyesinden ilgili miktarı çekme işlemi yapar. 3D secure başlatılması için session açar ve sessionId döner. Bu servisten dönen threeDsessionId iframe yöntemiyle 3Dyönlendirme servis url'ine eklenerek 3D doğrulama akışının yapılması sağlanır. Eğer 3D doğrulama başarılı ise Sistem otomatik olarak ödemeye gönderir. Bu method çağırıldıktan sonra kart bilgilerini ProcessCardForm methodu ile veya Ortak ödeme sayfamız ile kart bilgilerini iletebilirsiniz.

Taksitli / Taksitsiz işlem yapılabilir.

Method Name : ThreeDPayment

Http Method : Post

Content-Type : application/json

Request Parametreleri

Alan Adı	Tipi	Max Uzunluk	Zorunlu	Açıklama
clientId	long	20	Evet	Mağaza için belirlenmiş benzersiz numara
apiUser	string	100	Evet	Api kullanıcı adı
Rnd	string	24	Evet	İşlem için üretilmiş random numara
timeSpan	string	14	Evet	İşlem tarihi yyyyMMddHHmmss formatında
Hash	string	1024	Evet	Güvenlik kontrolünde kullanılacak hash stringi, bir hash bir kez kullanılacaktır.
callbackUrl	string	1024	Evet	3D sonucunun dönüleceği url
orderId	string	20	Opsiyonel	Sipariş Numarasıdır. Belirlenmediği takdirde sistem otomatik üretecektir.
Amount	long	19	Evet	İşlem Tutarı, son iki hane kuruştur. 1522 = 15 TL 22 Kuruş
Currency	int	3	Evet	İşlem Para birimi 949
installmentCount	int	2	Opsiyonel	Default "0"

Response Parametreleri

Alan Adı	Tipi	Max Uzunluk	Açıklama
Code	int	7	0 ise işlem başarılıdır. Diğer numaralar hatalıdır.
Message	string	256	İşleme ait mesajdır
threeDsessionId	string	256	3D işlem için üretilen Unique numara

Request	Response
<pre>{ "clientId": 1, "apiUser": "testUser", "rnd": "****", "timeSpan": "20191121160000", "hash": "****", "orderId": "12345678901234567890", "callbackUrl": "****merchantCallbackUrl****", "amount": 100, "currency": 949, "installmentCount": 0 }</pre>	<pre>{ "code": 0, "message": "Başarılı", "threeDSessionId": "****" }</pre>

3d ile Ödeme için Json Format Örneği;

```
{
  "orderId": "string",
  "amount": 0,
  "currency": 0,
  "installmentCount": 0,
  "callbackUrl": "string",
  "description": "string",
  "clientId": 0,
  "apiUser": "string",
  "rnd": "string",
  "timeSpan": "string",
  "hash": "string"
}
```

4 Ödeme Sorgulama

PTT Akıllı Esnaf ta kaydı oluşturulan bir işlemin detayları sorgulanabilir. İşlem sorgulama servisidir. OrderID ile işlem sorgulama yapılabilmektedir.

Method Name : inquiry

Http Method : Post

Content-Type : application/json

Request Parametreleri

Alan Adı	Tipi	Max Uzunluk	Zorunlu	Açıklama
clientId	long	20	Evet	Mağaza için belirlenmiş benzersiz numara
apiUser	string	100	Evet	Api kullanıcı adı
Rnd	string	24	Evet	İşlem için üretilmiş random numara
timeSpan	string	14	Evet	İşlem tarihi yyyyMMddHHmmss formatında
Hash	string	1024	Evet	Güvenlik kontrolünde kullanılacak hash stringi, bir hash bir kez kullanılacaktır.
orderId	string	20	Evet	Ön Otorizasyon (PreAuth) işleminde gönderilen Sipariş numarasıdır.

Response Parametreleri

Alan Adı	Tipi	Max Uzunluk	Açıklama
Code	int	7	0 ise işlem başarılıdır. Diğer numaralar hatalıdır.
Message	string	256	İşleme ait mesajdır
Payments	list		İşleme ait ödeme listesini döner.

Request	Response
<pre>{ "clientId": 1, "apiUser": "testUser", "rnd": "***", "timeSpan": "20191121160000", "hash": "***", "orderId": "12345678901234567890" }</pre>	<pre>{ "code": 0, "message": "Başarılı", "payments": [{ "createDate": "20151012112230", "orderId": "12345678901234567890", "code": 0, "message": "Success", "bankCode": 111, "amount": 22.11, "currency": "949", "clientId": 1 }] }</pre>

	}] }
--	-------------

5 İptal

Bankalar gün sonu almadan önce işlemin iptal edilmesidir. OrderId ile ödeme işleminin iptal edilmesini sağlar.

Method Name : void

Http Method : Post

Content-Type : application/json

Request Parametreleri

Alan Adı	Tipi	Max Uzunluk	Zorunlu	Açıklama
clientId	long	20	Evet	Mağaza için belirlenmiş benzersiz numara
apiUser	string	100	Evet	Api kullanıcı adı
Rnd	string	24	Evet	İşlem için üretilmiş random numara
timeSpan	string	14	Evet	İşlem tarihi yyyyMMddHHmmss formatında
Hash	string	1024	Evet	Güvenlik kontrolünde kullanılacak hash stringi, bir hash bir kez kullanılacaktır.
orderId	string	20	Evet	Orijinal İşleme Ait Sipariş Numarasıdır.

Response Parametreleri

Alan Adı	Tipi	Max Uzunluk	Açıklama
Code	int	7	0 ise İşlem başarılıdır. Diğer numaralar hatalıdır.
Message	string	256	İşleme ait mesajdır
orderId	string	20	Sipariş Numarasıdır. Belirlenmediği takdirde sistem otomatik üretecektir.

Request	Response
{ "clientId": 1,	{ "code": 0,

<pre>"apiUser": "testUser", "rnd": "****", "timeSpan": "20191121160000", "hash": "****", "orderId": "12345678901234567890" }</pre>	<pre>"message": "Başarılı", "orderId": "12345678901234567890", }</pre>
--	--

6 İade

Bankalar gün sonu aldıktan sonra işlemin kısmi veya tam olarak iade edilmesidir. Ödeme işlemi iade eder. Tam iade ve kısmi iade yapılabilmesini sağlar.

Method Name : refund

Http Method : Post

Content-Type : application/json

Request Parametreleri

Alan Adı	Tipi	Max Uzunluk	Zorunlu	Açıklama
clientId	long	20	Evet	Mağaza için belirlenmiş benzersiz numara
apiUser	string	100	Evet	Api kullanıcı adı
Rnd	string	24	Evet	İşlem için üretilmiş random numara
timeSpan	string	14	Evet	İşlem tarihi yyyyMMddHHmmss formatında
Hash	string	1024	Evet	Güvenlik kontrolünde kullanılacak hash stringi, bir hash bir kez kullanılacaktır.
orderId	string	20	Evet	Orijinal İşleme Ait Sipariş Numarasıdır.
Amount	long	19	Evet	İade İşlem Tutarı, son iki hane kuruştur. 1522 = 15 TL 22 Kuruş

Response Parametreleri

Alan Adı	Tipi	Max Uzunluk	Açıklama
Code	int	7	0 ise işlem başarılıdır. Diğer numaralar hatalıdır.
Message	string	256	İşleme ait mesajdır
orderId	string	20	Sipariş Numarasıdır.

Request	Response
---------	----------

<pre>{ "clientId": 1, "apiUser": "testUser", "rnd": "****", "timeSpan": "20191121160000", "hash": "****", "orderId": "12345678901234567890", "amount": 2211 }</pre>	<pre>{ "code": 0, "message": "Başarılı", "orderId": "12345678901234567890", }</pre>
---	---

7 İşlem Listeleme

Tarihe göre işlemleri listeler. Tarih gönderilerek iletilen tarihteki tüm işlemlerin listelenmesini sağlar.

Method Name : history
Http Method : Post
Content-Type : application/json

Request Parametreleri

Alan Adı	Tipi	Max Uzunluk	Zorunlu	Açıklama
clientId	long	20	Evet	Mağaza için belirlenmiş benzersiz numara
apiUser	string	100	Evet	Api kullanıcı adı
Rnd	string	24	Evet	İşlem için üretilmiş random numara
timeSpan	string	14	Evet	İşlem tarihi yyyyMMddHHmmss formatında
Hash	string	1024	Evet	Güvenlik kontrolünde kullanılacak hash stringi, bir hash bir kez kullanılacaktır.
reconciliationDate	int	8	Evet	İşlem yapılan tarihtir. yyyyMMdd
Page	int	inx max	Evet	Paging yapısı gereği kullanılan Page Numarası ilk sayfa için 1 , ikinci sayfa için 2...
pageSize	int	3	Evet	Paging yapısı gereği kullanılan Page Uzunluğu. 10 tane kayıt için 10.

Response Parametreleri

Alan Adı	Tipi	Max Uzunluk	Açıklama
Code	int	7	0 ise işlem başarılıdır. Diğer numaralar hatalıdır.
message	string	256	İşleme ait mesajdır

count	int	inx max	Filtreye uygun toplam kayıt sayısını döner.
payments	list		İşleme ait ödeme listesini döner.

Request	Response
<pre>{ "clientId": 1, "apiUser": "testUser", "rnd": "****", "timeSpan": "20191121160000", "hash": "****", "reconciliationDate": 20150101, "page": 1, "pageSize": 10 }</pre>	<pre>{ "code": 0, "message": "Başarılı", "payments": [{ "createDate": "20151012112230", "orderId": "12345678901234567890", "code": 0, "message": "Success", "bankCode": 111, "amount": 22.11, "currency": "949", "clientId": 1 }] }</pre>

8 Ortak Ödeme Sayfası

Method Name : threeDSecure

Server tarafından startThreeDSession servisi ile threeDSessionId alınarak client tarafına iletilir. client tarafında browser aracılığıyla threeDSecure url'ine get işlemi gerçekleştirilir. Parametre olarak threeDSessionId değerleri eklenir. threeDSessionId: 3D doğrulama işleminde kullanılan benzersiz değerdir.

Not: Kullanıcının bankanın 3D sayfasına yönlendirme yapılabilmesi için browser desteği gerekmektedir.

İşlem sonucunu threeDSessionResult servisinden de öğrenebilirsiniz.

Örnek :

9 ProcessCardForm

Ortak ödeme sayfasının kullanılmadığı durumda bu method çağırılacaktır. Ödeme işlemi başlatıldıktan sonra kart bilgilerinin iletileceği methoddur.

```
ThreeDSessionId=*****&  
CardHolderName=*****&  
CardNo=*****&  
ExpireDate=*****&  
Cvv=*****
```

10 Hash Mekanizması

```
private string CreateHash()  
{  
    var ApiPass = "";  
    var ClientId = "";  
    var ApiUser = "";  
    var Rnd = "";  
    var TimeSpan = "";  
    var hashString = ApiPass + ClientId + ApiUser + Rnd + TimeSpan;  
    System.Security.Cryptography.SHA512 sha = new  
    System.Security.Cryptography.SHA512CryptoServiceProvider();  
    byte[] bytes = Encoding.UTF8.GetBytes(hashString);  
    byte[] hashingbytes = sha.ComputeHash(bytes);  
    var hash = Convert.ToBase64String(hashingbytes);  
    return hash;  
}
```

11 Ortam ve Test Kart Bilgileri

Test Ortam Bilgileri

URL= <https://prepaeo.ptt.gov.tr/api/Payment>

ApiUser =Entegrasyon_01

ApiPass= gkk4l2*TY112

ClientID=1000000032

Test Kartları

Kart No	Vade Tarihi	CVV	3D Secure Şifre
4159560047417732	08/24	123	
4119790155203496	24/04	579	123456
4119790166544284	24/04	961	123456
4090700101174272	104	12/22	
4090700090840057	592	11/22	

“Test kurulumunuzu tamamladıktan sonra başvurunuz onaylanıp Üye İş Yeriniz açıldıktan sonra Prod (üretim) ortamı API key bilgileri paylaşılacaktır.”**

Prod Ortam Bilgileri

URL= <https://aeo.ptt.gov.tr/api/Payment>

ApiUser =Cep telefonunuza sms olarak iletilecektir.

ApiPass= Cep telefonunuza sms olarak iletilecektir.

ClientID= Cep telefonunuza sms olarak iletilecektir.

Method	Hata mesajı TR	Hata Mesajı EN
Void	throw new ApiException(MessageCodes.UnauthorizedOperationForMerchant ApiUser) --> UnauthorizedOperationForMerchantApiUser = 250	
	Tr için --> "Orjinal Kayıt Bulunamadı"	Eng için --> "Original Not Found"
	Tr için --> "Başarılı"	Eng için --> "Success"
	Tr için --> "Hatalı"	Eng için --> "Fail"
Refund	throw new ApiException(MessageCodes.UnauthorizedOperationForMerchant ApiUser) -->	
	Tr için --> "İade Tutarı Satış Tutarından Büyük Olamaz"	Eng için --> "Refund Amount Error"
	Tr için --> "Toplam İade Edilecek Tutar Satış Tutarından Büyük Olamaz"	Eng için --> "Refunded Amount Error"
	throw new ApiException(MessageCodes.MerchantNotFound) --> MerchantNotFound = 200	
	Tr için --> "Başarılı"	Eng için --> "Success"
	Tr için --> "Hatalı"	Eng için --> "Fail"
Inquiry	throw new ApiException(MessageCodes.UnauthorizedOperationForMerchant ApiUser) --> UnauthorizedOperationForMerchantApiUser = 250	
History	throw new ApiException(MessageCodes.UnauthorizedOperationForMerchant ApiUser) --> UnauthorizedOperationForMerchantApiUser = 250	
	Tr için --> "Başarılı"	Eng için --> "Success"
ThreeDPayment	throw new ApiException(MessageCodes.UnauthorizedOperationForMerchant ApiUser) --> UnauthorizedOperationForMerchantApiUser = 250	
	Tr için --> "Orjinal Kayıt Bulunamadı"	Eng için --> "Original Not Found"
	Tr için --> "Başarılı"	Eng için --> "Success"

12 Sıkça Sorulan Sorular

Sıkça sorulan sorular geldikçe bu başlık altına eklenecektir.

Soru - 1: Ödeme işlemini nasıl başlatacağım.

Cevap - 1: threeDPayment methodunu kullanarak ödeme isteğinizi başlatabilirsiniz.

Soru - 2: Kart bilgilerini nasıl göndereceğim

Cevap - 2: kendi uygulamanızda kart bilgilerini aldığınız bir sayfa var ise ProcessCardForm methodu ile kart bilgilerini gönderebilirsiniz. Kart bilgilerinizi aldığınız bir sayfa yok ise Ortak ödeme sayfasını kullanarak kart bilgilerini gönderebilirsiniz.

Soru - 3: Ödeme isteğini başlattım fakat response bilgisi boş dönüyor. Ne yapmalıyım.

Cevap – 3: Request bilgisinde gönderdiğiniz URL i kontrol ediniz. Sabit URL in sonuna method ismini eklemeniz gerekiyor. ÖR= <https://prepaeo.ptt.gov.tr/api/Payment/threeDPayment>

Soru - 4: <https://prepaeo.ptt.gov.tr/api/Payment/threeDPayment> servisi tarayıcı üzerinden çağırdığımda **Http/1.1 Service Unavailable** hatası alıyorum. Ne yapmalıyım?

Cevap – 4: Dış Ip adresinize yetki verilmedi anlamına gelmektedir. destek@akilliesnaf.ptt.gov.tr adresine Dış Ip adresinizi mail atarak erişim yetkisi isteyebilirsiniz. Dış Ip adresinizi öğrenmek için <https://whatismyipaddress.com/tr/ip-im> adresine girerek görebilirsiniz. Dış Ip adresinize yetki alabilmek için sabit bir IP adresine sahip olmanız gerekmektedir.