



# Sanal Pos Entegrasyon Dokümanı

---

Non3D Model

# İçindekiler Tablosu

Non3D Ödeme (3D'siz Ödeme).....	3
1. Banka Tarafından Sağlanacak Bilgiler.....	3
2. Entegrasyon İçin Kullanılan Diğer Bilgiler.....	3
3. Banka Tarafında Dönen Cevap Bilgileri .....	5
4. İşlemden Kullanılan Zorunlu Parametreler Ve Açıklamaları .....	5
5. Entegrasyon .....	6
5.1 Kullanılan Mesaj Yapıları .....	6
5.1.1. Ödeme İşlemi Mesaj Yapısı.....	6
5.1.2. Ödeme İşlem Sonucu Mesaj Yapısı .....	7
6. HashData Hesaplaması.....	8
7. Örnek Kod Parçaları .....	8
7.1. Ödeme İşlemi Yapılması.....	9
8. Non3D Ödeme Noktası Adresi .....	10
9. Test Ortam Bilgileri .....	10

## Non3D Ödeme (3D'siz Ödeme)

Kullanıcıların 3D Secure güvenlik sistemi olmadan daha hızlı ödeme yapmasına olanak sağlayan sistemdir. 3D Secure ödeme sistemlerinin sunduğu avantajları sağlamamaktadır.

Non3D sistemi uygulanan ödeme işlemlerinde kullanıcıdan kart bilgileri alındıktan sonra, ödeme işlemi yapılır. Ödeme işleminin gerçekleşebilmesi için kartın 3D Secure olmayan ödeme sistemlerine açık olması gerekmektedir.

Kuveyt Türk Katılım Bankası tarafından sunulan Sanal Pos servisi Non3D ödeme işlemlerini desteklemektedir. İşyerinin işlem alabilmesi için banka tarafında yetkilendirme yapılmaktadır.

Üye işyerinin Non3D ödeme uygulayabilmesi için aşağıdaki bilgilere sahip olması gerekmektedir.

### 1. Banka Tarafından Sağlanacak Bilgiler

**CustomerId (Müşteri Numarası):** Üye işyerinin Kuveyt Türk'teki Sanal Pos'un bağlı hesaba ait müşteri numarasıdır.

**MerchantId (Mağaza Numarası):** Üye işyerinin Kuveyt Türk Sanal Pos servisinde kayıtlı özel mağaza numarasıdır. Başvuru onay mailinde üye işyerine gönderilmektedir.

**PostURL (Ödeme Noktası Adresi):** Sanal Pos'un sunduğu ödeme yöntemine ait adreslerdir. Ödeme işlemine ait bilgiler bu adrese gönderilecektir. (Bkz. "[Non3D Secure Ödeme Noktası Adresi](#)")

### 2. Entegrasyon İçin Kullanılan Diğer Bilgiler

**APIVersion (Uygulama Versiyonu):** Uygulama versiyonunu ifade etmektedir. Sabit değer TDV2.0.0 gönderilebilmektedir.

**HashData (Doğrulama Anahtarı):** Ödeme işlemine ait bilgilerin doğruluğunun kontrol edilmesi için kullanılan veridir. Daha sonra sağlanacak algoritma ile üretilcek bu veri, Sanal Pos tarafında da üretilerek bu veriler karşılaştırılmaktadır. Bu kontrolden geçemeyen ödemeler gerçekleştirilmemektedir.

**CardNumber (Kart Numarası):** Ödemeyi yapacak kişiye ait kartın üzerindeki 16 haneli numaradır, 16 karakter olacak ve aralarında boşluk/sembol olmayacak şekilde gönderilmelidir.

**CardExpireDateYear (Kart Son Kullanım Yılı):** Kartın ön yüzünde bulunan son kullanma tarihine ait yıl bilgisi, 2 karakter olarak gönderilmelidir. (Örneğin; 2025 için 25)

**CardExpireDateMonth (Kart Son Kullanım Ayı):** Kartın ön yüzünde bulunan son kullanma tarihine ait ay bilgisi, 2 karakter olarak gönderilmelidir. (Örneğin; Ocak ayı için 01)

**CardCVV2:** Kartın arka yüzündeki 3 haneli güvenlik kodudur. 3 karakter olarak gönderilmelidir.

**CardHolderName (Kart Sahibinin Adı Soyadı)** : Kartın ön yüzünde yer alan adı ve soyadı bilgisi gönderilmelidir.

**CardType (Kart Türü)** : TROY, VISA veya MasterCard olarak gönderilmelidir. Default bir değer kullanılabilir. (Örneğin; VISA)

**BatchID (Günsonu Numarası)**: Satış işleminde sabit değer 0 gönderilmelidir.

**TransactionType (İşlem Türü)** : Sabit değer olarak "Sale" gönderilmelidir.

**InstallmentCount (Taksit Sayısı)**: Taksit sayısı bu alana girilmelidir, taksit yapma yetkiniz bulunmuyorsa 0 olarak gönderilmelidir. Tarım işyeri olanlar harcama öteleme yapacağı zaman DeferringCount (Öteleme Ay Sayısı) parametresini kullanmalıdır. Gerekli taksit tanımlamaları için şube ile görüşülmelidir.

**Amount (Tutar)**: İşlem tutarının girildiği alandır. Bu alanda hiçbir noktalama işareti kullanılmadan ve gerçek tutarın 100 katı gönderilmelidir. (Örneğin; 102.65 TL için 10265, 1 TL için 100)

**DisplayAmount (Tutar Görünümü)**: Tutar (Amount) değeri ile aynı olacak şekilde gönderilmelidir.

**CurrencyCode (Para Kodu)**: Para birimidir, "0949" olarak gönderilmesi gerekmektedir. Sanal posla yalnızca TL ile işlem alınmaktadır.

**MerchantOrderId (Üye İşyeri Sipariş Numarası)**: İşlemin üye işyeri tarafında yer alan numarasıdır. Bu alana işyeri istediği sabit veya değişken değer atayabilmektedir.

**TransactionSecurity (İşlem Güvenliği)**: İşlemin yapılacağı güvenlik türü bilgisidir. Non3DPaygate için 1 değeri gönderilmelidir.

**UserName (Api Kullanıcı Adı)**: <https://kurumsal.kuveytturk.com.tr> adresine login olarak kullanıcı işlemleri sayfasında API rolünde oluşturulan kullanıcı adıdır. (Yönetim - Kullanıcı İşlemleri- Kullanıcı Ekleme) Maksimum 10 karakter uzunluğunda, harf veya rakamdan oluşturulmalıdır.

**Password (Api Şifresi)**: <https://kurumsal.kuveytturk.com.tr> adresine login olarak kullanıcı işlemleri sayfasında oluşturulan API kullanıcısının şifre bilgisidir.

### 3. Banka Tarafında Dönen Cevap Bilgileri

**OrderId (İşlem Numarası):** İşlemin Sanal Pos sisteminde kayıtlı numarasıdır. Banka Tarafında üretilmektedir. İade ve iptal işlemlerinde kullanılmaktadır.

**ProvisionNumber (Provizyon Numarası):** Başarılı işlemlerde kart bankasının vermiş olduğu otorizasyon numarasıdır. Kart bankası tarafında üretilmektedir. İade ve iptal işlemlerinde kullanılmaktadır.

**RRN (Referans Numarası1):** Pos bankası tarafında üretilen işlem referans numarasıdır. İade ve iptal işlemlerinde kullanılmaktadır.

**Stan (Referans Numarası2):** Pos bankası tarafında üretilen işlem referans numarasıdır. İade ve iptal işlemlerinde kullanılmaktadır.

**ResponseCode (Cevap Kodu):** Genel kabul görmüş kart veya pos bankası cevabıdır. Başarılı işlemlerde "00" kodu verilmektedir. Her bir kod farklı bir cevabı bulunmaktadır.

**ResponseMessage (Cevap Mesajı):** Genel kabul görmüş kart veya pos bankası cevabının açıklamasıdır.

**Not:** Cevap kodu ve cevap mesajı listesi <https://kurumsal.kuveytturk.com.tr> adresindeki Yönetim - Dokümantasyon alanında paylaşımdadır.

### 4. İşlemden Kullanılan Zorunlu Parametreler Ve Açıklamaları

Bu bölümde üye işyeri ve Sanal Pos servisi arasında gerçekleştirilecek entegrasyon için bilgiler yer almaktadır. Üye işyerinin Non3D ödeme uygulayabilmesi için aşağıdaki bilgilere sahip olması gerekmektedir.

Alan Adı	Z: Zorunlu, O: Opsiyonel	Tip
KuveytTurkVPosMessage(Başlangıç Tag)		
APIVersion	O	
HashData	Z	
MerchantId	Z	Tamsayı
CustomerId	Z	Tamsayı
UserName	Z	Text
CardNumber	Z	Text
CardExpireDateYear	Z	Text
CardExpireDateMonth	Z	Text
CardCVV2	Z	Text
CardHolderName	Z	Text
CardType	O	Text
BatchID	O	Tamsayı
TransactionType	Z	Text
InstallmentCount	Z	Tamsayı
InstallmentMaturityCommisionFlag	O	Tamsayı

Amount	Z	Tamsayı
DisplayAmount	Z	Tamsayı
CurrencyCode	Z	Text
MerchantOrderId	O	Text
TransactionSecurity	Z	Tamsayı

## 5. Entegrasyon

Sanal Pos tarafından sunulan Non3D Secure ödeme yöntemi için kullanılacak örnek bir entegrasyon işlemi bu kısımda anlatılmaktadır.

Üye işyeri sipariş detayını görüntüler. Kullanıcıdan, Non3D Secure ödeme işlemi seçmesi beklenir. Üye işyeri kullanıcıdan kart bilgilerini alır. Kart, sipariş ve üye işyerine ait bilgileri içeren mesaj oluşturulur. Mesajda yer alacak HashData alanına dikkat edilmelidir. Bu mesaj yapısı XML formatındadır. Bir sonraki bölümde örnek mesaj yapısı yer almaktadır. (Bkz. "[Kullanılan Mesaj Yapıları](#)")

Üye işyeri oluşturduğu XML mesajı SanalPos tarafından sunulan Non3D Secure ödeme noktasına HTTP POST yöntemi ile gönderir. Bu işlem sonucunda XML yapısında bir cevap yer almaktadır. Üye işyeri aldığı bu cevabı işleyip işlemlerine devam etmelidir.(Bkz. "[Ödeme İşlemi Yapılması](#)", "[Sanal Pos Non3D Secure Ödeme Noktası Adresi](#)")

Gelen XML cevapta ResponseCode değeri "00" olduğu durumda ödeme işlemi gerçekleşmiş olmaktadır.

### 5.1 Kullanılan Mesaj Yapıları

#### 5.1.1. Ödeme İşlemi Mesaj Yapısı

```
<KuveytTurkVPosMessage xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
xmlns:xsd="http://www.w3.org/2001/XMLSchema">
<APIVersion>TDV2.0.0</APIVersion>
<HashData> OIV5gzIYnMcqXovR4/K37IFrCCE=</HashData>
<MerchantId>496</MerchantId>
<CustomerId>400235</CustomerId>
<UserName>apitest</UserName>
<CardNumber> </CardNumber>
<CardExpireDateYear>**</CardExpireDateYear>
<CardExpireDateMonth>**</CardExpireDateMonth>
<CardCVV2>**</CardCVV2>
<CardHolderName>test</CardHolderName>
<CardType>MasterCard</CardType>
<BatchID>0</BatchID>
<TransactionType>Sale</TransactionType>
<InstallmentCount>0</InstallmentCount>
<Amount>100</Amount>
<DisplayAmount>100</DisplayAmount>
<CurrencyCode>0949</CurrencyCode>
<MerchantOrderId> siparis_1</MerchantOrderId>
```

```
<TransactionSecurity>1</TransactionSecurity>
<TransactionSide>Sale</TransactionSide>
</KuveytTurkVPosMessage>
```

### 5.1.2. Ödeme İşlem Sonucu Mesaj Yapısı

```
<VPosTransactionResponseContract xmlns:xsd="http://www.w3.org/2001/XMLSchema"
xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance">
<VPosMessage>
<APIVersion>TDV2.0.0</APIVersion>
<HashData>uW9eqzWGeYeicgymA9Zo4g30Q/M=</HashData>
<MerchantId>496</MerchantId>
<SubMerchantId>0</SubMerchantId>
<CustomerId>400235</CustomerId>
<UserName>apitest</UserName>
<CardNumber>*****</CardNumber>
<CardHolderName>test</CardHolderName>
<CardType>MasterCard</CardType>
<BatchID>0</BatchID>
<TransactionType>Sale</TransactionType>
<InstallmentCount>0</InstallmentCount>
<Amount>100</Amount>
<CancelAmount>0</CancelAmount>
<DisplayAmount>100</DisplayAmount>
<MerchantOrderId>siparis_1</MerchantOrderId>
<FECAmount>0</FECAmount>
<CurrencyCode>0949</CurrencyCode>
<QeryId>0</QeryId>
<DebtId>0</DebtId>
<SurchargeAmount>0</SurchargeAmount>
<SGKDebtAmount>0</SGKDebtAmount>
<TransactionSecurity>1</TransactionSecurity>
<TransactionSide>Auto</TransactionSide>
<DeferringCount xsi:nil="true" />
<InstallmentMaturityCommisionFlag>0</InstallmentMaturityCommisionFlag>
<PaymentId xsi:nil="true" />
<ReferencId>611a5de98ef94070a42d5876948df38d</ReferencId><OrderPOSTransactionId
xsi:nil="true" /><TranDate xsi:nil="true" />
<EntryGateMethod>VPOS_Non3DPayGate</EntryGateMethod>
<TransactionUserId xsi:nil="true" />
</VPosMessage>
<IsEnrolled>true</IsEnrolled>
<IsVirtual>>false</IsVirtual>
<ProvisionNumber>036840</ProvisionNumber>
<RRN>035719459011</RRN><Stan>459011</Stan>
<ResponseCode>00</ResponseCode>
```

```
<ResponseMessage>OTORİZASYON VERİLDİ</ResponseMessage>
<OrderId>40790262</OrderId>
<TransactionTime>2020-12-22T19:26:20.7889455+03:00</TransactionTime>
<MerchantOrderId>siparis_1</MerchantOrderId>
<HashData>4mJBGfRxtmlh1Lfj5wEqx/GRnSc=</HashData>
<ReferenceId>611a5de98ef94070a42d5876948df38d</ReferenceId>
<BusinessKey>202012221020904998840000001</BusinessKey>
</VPosTransactionResponseContract>
```

**Not:** Ödeme işlemine ait cevap mesajının içerisinde yer alan VPosMessage alanında, üye işyerlerinin ödeme işlemi için gönderdikleri mesaj yer almaktadır.

## 6. HashData Hesaplaması

Gelen cevabın Kuveyt Türk Sanal POS tarafından gönderildiğinin kontrol edilebilmesi için HashData değeri yer almaktadır. HashData alanının, üye işyerinin kendi oluşturduğu HashData ile aynı değere sahip olması bu cevap mesajının Kuveyt Türk Sanal POS tarafından gönderildiğini doğrulanmıştır.

Hash hesaplamasında kullanılan parametreler şöyledir: MerchantId, MerchantOrderId, Amount, OkUrl, FailUrl, UserName, hashPassword

**Not:** Okurl ve Failurl alanları, kullanmadığınız durumlarda Hash hesaplamasına dahil edilmemelidir.

## 7. Örnek Kod Parçaları

```
public static String CreateHashString(KuveytTurkVPosMessage message, string hashPassword)
{
    string newHash = message.MerchantId + message.MerchantOrderId +
    message.Amount.ToAmountString() + message.OkUrl + message.FailUrl + message.UserName +
    hashPassword;
    return newHash;
}
```

```
public static String ComputeHash(String hashstr)
{
    System.Security.Cryptography.SHA1 sha = new
    System.Security.Cryptography.SHA1CryptoServiceProvider();
    byte[] hashbytes = System.Text.Encoding.GetEncoding("ISO-8859-9").GetBytes(hashstr);
    byte[] inputbytes = sha.ComputeHash(hashbytes);
    String hash = Convert.ToBase64String(inputbytes);
    return hash;
}
```

```
private static string DataPost(string uriToPost, string dataToPost)
{
    byte[] buffer = Encoding.UTF8.GetBytes(dataToPost);
    HttpWebRequest WebReq = (HttpWebRequest)WebRequest.Create(uriToPost);
    WebReq.Timeout = 5 * 60 * 1000;
}
```



```

WebReq.Method = "POST";
WebReq.ContentType = "application/xml";
WebReq.ContentLength = buffer.Length;
WebReq.CookieContainer = new CookieContainer();
Stream ReqStream = WebReq.GetRequestStream();
ReqStream.Write(buffer, 0, buffer.Length);
ReqStream.Close();
WebResponse WebRes = WebReq.GetResponse();
Stream ResStream = WebRes.GetResponseStream();
StreamReader ResReader = new StreamReader(ResStream);
string responseString = ResReader.ReadToEnd();
return responseString;
}

```

## 7.1. Ödeme İşlemi Yapılması

```

[HttpPost]
public string Payment(VPosTransactionResponseContract model)
{
    ViewBag.Message = "";
    KuveytTurkVPosMessage request = new KuveytTurkVPosMessage();
    request.TransactionType = "Sale";
    request.CardNumber = model.VPosMessage.CardNumber;
    request.CardExpireDateMonth = model.VPosMessage.CardExpireDateMonth;
    request.CardExpireDateYear = model.VPosMessage.CardExpireDateYear;
    request.CardType = model.VPosMessage.CardType;
    request.CardCVV2 = model.VPosMessage.CardCVV2;
    request.CardHolderName = model.VPosMessage.CardHolderName;
    request.CurrencyCode = "0949";
    request.APIVersion = "TDV2.0.0";
    request.InstallmentCount = "0";
    request.MerchantId = merchantId;
    request.CustomerId = customerId;
    request.UserName = userName;
    string password = hashedPassword;
    request.MerchantOrderId = model.VPosMessage.MerchantOrderId;
    request.Amount = model.VPosMessage.Amount;
    request.DisplayAmount = request.Amount.ToString();
    request.TransactionSecurity = model.VPosMessage.TransactionSecurity;
    string newHash = Helper.CreateHashString(request, password);
    request.HashData = Helper.ComputeHash(newHash);
    XmlSerializer x = new XmlSerializer(request.GetType());
    StringWriter sw = new StringWriter();
    x.Serialize(sw, request);
    XsdDataContractExporter exporter = new XsdDataContractExporter();
    string str = sw.ToString();
    str = exporter.Export(str);
    str = str.Replace("encoding=\"utf-16\"", "encoding=\"ISO-8859-1\"");
    string result = DataPost(Non3DPayGate Url, str);
    if (String.IsNullOrEmpty(result))
        result = "Boş Cevap";
    return result;
}

```

## 8. Non3D Ödeme Noktası Adresi

SanalPos Non3D Secure Ödeme Noktası Adresi (Production):

<https://sanalpos.kuveytturk.com.tr/ServiceGateWay/Home/Non3DPayGate>

SanalPos Non3D Secure Ödeme Noktası Adresi (Test):

<https://boatest.kuveytturk.com.tr/boa.virtualpos.services/Home/Non3DPayGate>

## 9. Test Ortam Bilgileri

### **Test Bilgileri**

CustomerId = "97228291"; // Müşteri Numarası  
MerchantId = "57902" ; // Mağaza Numarası

### **Api Bilgileri**

UserName="TEPKVT2021";  
Password="api123";

### **Test Kart Bilgileri**

Kart No: # 5188'9619'3919'2544#  
CVV: 929  
Expirydate: 06/25

**Not:** Test ortamında alınacak herhangi bir hatada bankaya bilgi vermelisiniz.

**Mail adresi:** [sanalposdestek@kuveytturk.com.tr](mailto:sanalposdestek@kuveytturk.com.tr)