

Version

1.4

*Pay***ten**

NestPay[®]

Merchant Integration 3D

www.payten.com

Document Information

Project/Product Name			
Project Manager			
Document Version No			
Document Code			
Prepared By		Preparation Date	03/03/2025
Reviewed By		Review Date	03/03/2025

Distribution List

From	Date	Phone/Fax

To	Action*	Due Date	Phone/Fax

* Action Types: Approve, Review, Inform, File, Action Required, Attend Meeting, Other (please specify)

Version History

Ver. No.	Ver. Date	Revised By	Description
1.1	04/05/2012	Payten	Bankadan dönen Hash parametresi için önemli not eklendi.
1.2	24 /04/2014	Payten	İsteğe bağlı alanlarda değişiklik yapıldı.
1.3	31/01/2024	Payten	Hash algoritması güncelleme
1.4	28/02/2025	Payten	Cors policy güncelleme

Proprietary Notice

The information contained in all sheets of this document, proposal, or quotation constitutes trade secrets and/or information that is commercial or financial and is deemed confidential or privileged. It is furnished to prospective customer in confidence with the understanding that prospective customer will not, without the permission of Payten Teknoloji A.Ş. (from now on called Payten), use or disclose for other than evaluation purposes the information contained herein which is solely confidential and proprietary to Payten ("**Payten Confidential Information**"). In the event a contract is awarded on the basis of this document, proposal, or quotation, prospective customer shall have the right to use and disclose Payten Confidential Information to the extent provided in the contract. The restriction does not limit prospective customer right to use or disclose such information if obtained from another source without restriction.

Contents

3D Model	5
Payten 3D Model Akışı.....	6
1.Hızlı Başlangıç Kılavuzu.....	6
1.1 Gönderilen Gizli Parametreler	7
1.2 Zorunlu Parametre Seti ile HTTP Örneği	7
1.3 3D Doğrulama.....	8
1.4 Ödeme	8
2. Temek Entegrasyonlar	11
2.1 HTTP Post Entegrasyonu	11
2.1.1 HTTP Formunda Zorunlu ve İsteğe Bağlı Parametrelere Örnek	11
2.2 Kart İşlemleri	12
2.2.1 MPI Yanıt Parametreleri	12
2.2.2 Olası mdStatus değerleri	12
2.2.3 API isteği oluşturma	13
2.3 Hash Kontrolü	13
3. Kod Örnekleri	14
3.1 ASP Kod Örneği	14
3.2 .Net C# Kod Örneği	16
3.3 .Net - VB Kod Örneği	18
3.4 JSP Kod Örneği.....	21
3.5 PHP Kod Örneği	23
APPENDIX A: Geçit Parametreleri.....	25
Zorunlu Parametreler	25
İsteğe Bağlı Parametreler	26
İşlem Yanıt Parametreleri	27
MPI Yanıt Parametreleri	28

3D Model

3D Model, 3D işlemleri destekleyen, ödeme sayfasına temel internet entegrasyon modelidir.

Temel Özellikler:

- 3D ile, Kredi Kartı ile yapılan işlemlerin güvenliğini sağlar.
- Üye iş yeri entegrasyonu için HTTP Post metodunu kullanır.
- Kredi kartı sayfası üye iş yeri tarafında tutulur.
- Kredi kartı otorizasyon işlemlerini üye iş yeri, API kullanarak yapar.

Gerekli olan tüm alışveriş verileri müşteriden sağlandıktan sonra, üye iş yeri sunucusunda benzersiz bir Sipariş Numarası oluşturulur. Bu sipariş numarası ve gerekli parametler (Sipariş tutarı, para birimi, müşteri adı/soyadı gibi) Nestpay' e (ödeme geçidi) HTTP Post metodu ile gönderilir.

Kart ödeme metotlarında üye iş yeri sunucusu kart detaylarını yani kart numarası, kart güvenlik numarasını ve son kullanma tarihi gibi bilgileri ibraz etmesi gerekir. Kart bilgileri müşteriden elde edildikten ve Sipariş verildikten sonra 3D akışı (kayıt ve doğrulama sorgulaması) başlar. 3D akışında, müşterinin 3D doğrulama bilgileri kartı veren banka tarafından doğrulanır. Bankanın müşteriye doğrulamak için kullandığı yöntemler değişiklik gösterebilir. Örneğin; 3D doğrulama metodunun kullanımında bankaya göre 3D güvenlik şifresi, tek kullanımlık şifre, güvenlik sorusu gibi yöntemlerle farklılık gösterir.

1. Kart bilgileri Nestpay tarafından alındığı için, müşteri bu bilgilerin üye iş yeri tarafından kaydedilmediğini bilir.
2. Entegrasyon süreci kolaydır.
3. Bankanın SSL sertifikası kullanılır. Bu nedenle, yazılımda modifikasyon yapılamaz.
4. Üye iş yeri zorunlu olan bilgiler dışında kendi istediği verileri de POST edebilir ve bu bilgileri bankadan geri alır. Örneğin; kullanıcı adı, kullanıcı e-posta adresi, kullanıcı kimlik numarası gibi.

1.1 Gönderilen Gizli Parametreler

Zorunlu girdi parametrelerini gizli olarak Nestpay Ödeme Geçidinde yer alan

https://[host_name]/fim/est3dgate linkine iletilir

clientid: Üye iş yeri numarası (Payten tarafından verilen numaradır.)

storetype: "3d_pay_hosting" (Mağaza da kullanılan ödeme model tipi)

hash: Kullanıcı doğrulama için Hash değeri

hashalgorithm: Kullanıcı doğrulama için kullanılan Hash algoritması

islemtipi: "Auth" (Satış)

amount: Tutar

currency: ISO para birimi kodu (TL için 949)

oid: Sipariş numarası

okUrl: Payten Ödeme Geçidine gelen başarılı işlem bilgilendirmesini üye iş yeri tarafındaki başarılı işlem bildirimi için önceden belirlenmiş olan URL' e iletilir.

failUrl: Payten Ödeme Geçidine gelen başarısız işlem bilgilendirmesini üye iş yeri tarafındaki başarısız işlem bildirimi için önceden belirlenmiş olan URL' e iletilir.

lang: Payten ödeme sayfasında kullanılan dil. (Türkçe için "tr", İngilizce için "en")

pan: Kredi Kartı Numarası

Ecom_Payment_Card_ExpDate_Year: Kredi kartı son kullanma tarihi (Yıl)

Ecom_Payment_Card_ExpDate_Month: Son kullanma tarihi (Ay)

1.2 Zorunlu Parametre Seti ile HTTP Örneği

```
<form method="post" action="https://host/fim/est3dgate">
  <input type="hidden" name="clientid" value="9900000000000001"/>
  <input type="hidden" name="storetype" value="3d" />
  <input type="hidden" name="hash" value="iej6cPOjDd4IKqXWQEznXWqLzLI=" />      <input
type="hidden" name="hashAlgorithm" value="ver3" />
  <input type="hidden" name="islemtipi" value="Auth" />
<input type="hidden" name="amount" value="91.96" />
<input type="hidden" name="currency" value="949" /> <input
type="hidden" name="oid" value="1291899411421" />
<input type="hidden" name="okUrl" value="https://www.teststore.com/success.php"/> <input
type="hidden" name="failUrl" value="https://www.teststore.com/fail.php"/> <input type="hidden"
name="lang" value="en" />
<input type="hidden" name="rnd" value="asdf" />
<input type="hidden" input name="pan" value="4242424242424242">
```

```
<input type="hidden" input name="Ecom_Payment_Card_ExpDate_Year" value="28" > <input  
type="hidden" input name="Ecom_Payment_Card_ExpDate_Month" value="10"></form>
```

1.3 3D Doğrulama

3D akışında, müşterinin 3D doğrulama bilgileri kartı veren banka tarafından doğrulanır. Bankanın müşteriyi doğrulamak için kullandığı yöntemler değişiklik gösterebilir. Örneğin; 3D doğrulama metodunun kullanımında bankaya göre 3D güvenlik şifresi, tek kullanımlık şifre, güvenlik sorusu gibi yöntemlerle farklılık gösterir.

1.4 Ödeme

MPI yanıt parametreleri, üye iş yerinden gelen tüm parametreler ve 3D doğrulama sonucu üye iş yerine gönderilir. MPI yanıt parametrelerinin tam listesi için bakınız: Ek A.

Doğrulanmış Full 3D İşlem için Temel MPI Yanıt Parametreleri:

Response: "Approved", "Declined" or "Error" (Yanıt Parametresi)
mdStatus: "1" 3D işlem durum kodu
eci: Elektronik Ticaret Göstergesi (Electronic Commerce Indicator)
cavv: Kart güvenlik numarası (Cardholder Authentication Verification Value, ACS tarafından belirlenir.)
md: Kart numarası yerine Hash
mdErrorMsg: "Authenticated" MPI hata mesajları
xid: Benzersiz internet işlem numarası

API ile ödeme yapmak için, gerekli API istek mesajı oluşturulur ve bu mesaj Nestpay API sunucusuna gönderilir (<https://host/fim/cc5xml>). (API özellikleri için lütfen "Nestpay Üye İş Yeri Entegrasyon API" dokümanına bakınız). İşlem sürece girer ve API' den yanıt mesajı gelir. Bu işlem yanıt parametrelerini içerir.

API isteğinde aşağıdaki alanlar doğru gönderilmelidir.

<Number>.
<PayerAuthenticationCode>.
<PayerSecurityLevel>
<PayerTxnId>

- **mdStatus = 1 (Full 3D), 2, 3 or 4 (Half 3D) durumunda:**

Number, PayerAuthenticationCode, PayerSecurityLevel ve PayerTxnId API isteği içinde gönderilmesi gerekir. Kart güvenlik numarası (CVV2) ve son kullanma tarihi (Expiry date) gönderilmez. MPI yanıt parametrelerinde kredi kartı numarası "md" parametresinde gönderilir.

- **mdStatus = 5, 7, 8 (MPI fallback)**

PayerAuthenticationCode, PayerSecurityLevel ve PayerTxnId değerlerinin gönderilmesine gerek yoktur. Standart 3D olmayan API isteği oluşturulur.(CVV2 ve son kullanım tarihi gönderilir, gerçek kredi kartı numarası kullanılır.)

- **mdStatus = 0 (doğrulama başarısız), 6 (ödeme yapılamadı.)**

<Number> elamanı için "md" parametre gönderilmesi gerekir <PayerAuthenticationCode> elamanı için "cavv" parametre gönderilmesi gerekir <PayerSecurityLevel> elamanı için "eci" parametre gönderilmesi gerekir <PayerTxnId> elamanı için "xid" parametre gönderilmesi gerekir.

3D Doğrulama Sonucu Dönen xid, cavv, eci ve md Değerlerinin Api ile Gönderilmesi

Api' ye gönderilen parametreler içinde kart bilgisi açık olarak yazılmaz. Kart numarası yerine MPI yanıt parametrelerinden md parametresinin değeri kart numarası yerine gönderilir.

```
<Number>540667:AD3547B4208D960B3A24422F47166B5C37BF8AE531FD43C2AB04381F34B7662B:3128:#</Number>
```

Full ve half 3D işlemler için Api isteğinde son kullanma tarihi (Expires), Cvv2 numarası (Cvv2Val), toplam tutar (Total) alanları gönderilmez;

```
<Expires>***</Expires>
<Cvv2Val>***</Cvv2Val>
<Total>9.95</Total>
```

Üye işyeri tutarlar gönderse bile dikkate alınmayacaktır, hatalı tutarlar reddedilecektir.

3D sorgusunda tutar olarak ne gönderiliyor ise API' ye o tutar post edilmeli. Çünkü son kullanıcı o tutar ile kendi bankasında doğrulama yapıyor ve o tutarın ödeneceğini bilir.

Ödeme işlemini üye iş yeri API' ye post edeceği için istenilen tutarı son kullanıcıdan çekmemesi için 3D sorgusundaki tutar baz alınır. Farklı tutar gönderilse dahi 3D sorgusunda gönderilen tutar çekilebilmesi için hata verilmez.

Örnek:

3D tutarı : 330,30 €

API'ye iletilen tutar : 453,60 €

Bu durumda 330,30 luk tutar bankaya provizyona gönderilecektir.

1.4.1.1 API ödeme isteği:

```
<?xml version="1.0" encoding="ISO-88599">
<CC5Request>
  <Name>apiuser</Name>
  <Password>apipassword</Password>
  <clientid>9900000000000001</clientid>
  <IPAddress>192.169.0.76</IPAddress>
  <oid>1292492233096</oid>
  <Type>Auth</Type>
  <Number>401550:B4DD659706E55C0FDB6AE46C4746C5B368577052FAF67A6901AF6342EED6DA3B:43
    51:#
  </Number>
  <amount>129,96</amount>
  <currency>949</currency>
  <PayerTxnId>IbjqznvVmFoAf99bRkOyegKQCY=</PayerTxnId>
  <PayerSecurityLevel>05</PayerSecurityLevel>
  <PayerAuthenticationCode> AAABBgIBegAAAAABcwESAAAAAA= </PayerAuthenticationCode>
</CC5Request>
```

Dikkat: Bu XML de yer alan ad ve şifre elementleri üye iş yeri adı ve şifresi değildir. Üye iş yerinin API kullanıcı bilgileri, üye iş yeri yöneticisinin yönetim ekranından oluşturduğu ad ve şifrelerdir. API kullanıcısı oluşturmak için lütfen Üye iş yeri Kullanıcı dokümanına bakınız.

Yanıt mesajında Response = "Approved" veya ProcReturnCode= "00" ise işlemin onaylandığını gösterir.

1.4.1.2 Cevap Örnek Yapısı:

```
<?xml version="1.0" encoding="ISO-8859-2"?>
<CC5Response>
```

```
<oid>1292492722637</oid>
<GroupId>1292492722637</GroupId>
<Response>Approved</Response>
<AuthCode>801022</AuthCode>
<HostRefNum>035010000340</HostRefNum>
<ProcReturnCode>00</ProcReturnCode>
<TransId>103501145290410086</TransId>
<ErrMsg></ErrMsg>
<Extra>
  <SETTLEID>20101216</SETTLEID>
  <TRXDATE>20101216 11:45:29</TRXDATE>
  <ERRORCODE></ERRORCODE>
  <NUMCODE>00</NUMCODE>
  <CARDBRAND>VISA</CARDBRAND>
</Extra>
</CC5Response>
```

2. Temek Entegrasyonlar

2.1 HTTP Post Entegrasyonu

Geçerli bir sipariş parametresi HTTP form ile gizli parametre olarak Nestpay' e gönderilir.

28 byte uzunluğundaki base-64 kodlu xid parametresi, 3D güvenli işlemler için gerekli olan benzersiz İnternet işlem kimliğidir. İşyeri tarafından gönderilmediği takdirde sistem tarafından otomatik olarak oluşturulacaktır.

2.1.1 HTTP Formunda Zorunlu ve İsteğe Bağlı Parametrelere Örnek

```
<form method="post" action="https://host/fim/Nestpaygate">
  <input type="hidden" name="clientid" value="9900000000000001"/>
  <input type="hidden" name="storetype" value="3d" />
  <input type="hidden" name="hash" value="iej6cPOjDd4IKqXWQEznXWqLzLI=" />
  <input type="hidden" name="trantype" value="Auth" />
  <input type="hidden" name="amount" value="91.96" />
  <input type="hidden" name="currency" value="949" />
  <input type="hidden" name="instalment" value="" />
  <input type="hidden" name="oid" value="1291899411421" />
  <input type="hidden" name="okUrl" value="https://www.teststore.com/success.php" />
  <input type="hidden" name="failUrl" value="https://www.teststore.com/fail.php" />
  <input type="hidden" name="lang" value="tr" />
```

```
<input type="hidden" name="rnd" value="asdf" />
<input type="hidden" input name="pan" value="4242424242424242">
<input type="hidden" input name="Ecom_Payment_Card_ExpDate_Year" value="28" >
<input type="hidden" input name="Ecom_Payment_Card_ExpDate_Month" value="10">
<input type="hidden" input name="cv2" value="000">
<input type="hidden" input name="encoding" value="utf-8">
</form>
```

2.2 Kart İşlemleri

Kart bilgileri onaylandıktan sonra müşteri ile 3D doğrulama akışı başlar. 3D doğrulama süreci tamamlandıktan sonra MPI yanıt parametreleri ve üye iş yeri tarafından gönderilmiş olan tüm parametreler ödeme yapabilmesi için yine üye iş yerine geri gönderilir. mdStatus alanında 3D güvenli işlem durum kodu görüldüğünde ödeme tamamlanmış olur.

2.2.1 MPI Yanıt Parametreleri

mdStatus: 3D işlem durum kodu

txstatus: Arşiv için 3D durumu

eci: Elektronik Ticaret Göstergesi (Electronic Commerce Indicator)

cavv: Kart güvenlik numarası (Cardholder Authentication Verification Value, ACS tarafından belirlenir.)

md: Kart numarası yerine Hash

mdErrorMsg: MPI hata mesajları

2.2.2 Olası mdStatus değerleri

- 1 = Doğrulanmış işlem (Full 3D)
- 2, 3, 4 = Kart kayıtlı değil (Half 3D)
- 5, 6, 7, 8 = Geçerli doğrulama yok veya sistem hatası
- 0 = Doğrulama başarısız.

mdStatus {1,2,3,4,5,7,8} değerleri almış ise: ödeme Nestpay API tarafından tamamlanmış anlamına gelir.

mdStatus {0,6} ise: diğer tüm işlemler ret edilmiş ve hatalı işlem olarak işaretlenmiş anlamına gelir.

mdStatus {5,7,8} ise: sanal POS sağlayan bankaya 3D olmayan işlem olarak döner.

2.2.3 API isteği oluşturma

3D modelde; CVV2 ve son kullanma tarihi API isteği içerisinde gönderilmez. API isteğinde gönderilmesi gereken 3D işlem detayları doğru şekilde ayarlanarak API istek mesajına gönderilir. Bunlar:

<Number> elamanı için "md" parametre ayarlanması gerekir
<PayerAuthenticationCode> elamanı için "cavv" parametre ayarlanması gerekir
<PayerSecurityLevel> elamanı için "eci" parametre ayarlanması gerekir
<PayerTxnId> elamanı için "xid" parametre ayarlanması gerekir

Bazı durumlarda (mdStatus=2) "eci" boş olabilir. Bu durumda dahi PayerSecurityLevel boş dizge olarak ayarlanır ve gönderilmesi gerekir.

API istekleri <https://host/fim/api> adresine gönderilir. API özellikleri için lütfen API Entegrasyon Dokümanına bakınız.

2.2.3.1 Olası API İşlem Sonuçları

- Yanıt: "Approved" (Onaylanan)
ProcReturnCode "00" olur. Bu, işlemin doğrulanmış olduğunu gösterir.
- Yanıt: "Declined" (Reddedilen)
ProcReturnCode "00" ve "99" dan başka, sanal POS sağlayan banka 2 basamaklı farklı bir sayı ile hata kodu alır. Bu mesajda sanal POS sağlayan bankanın işleme onay vermediği anlaşılır. ErrMsg parametreleri hatanın detay açıklamalarını verir. Sanal POS sağlayan bankadan gelen hata kodlarının detaylı açıklamalarında Ek B.' deki ProcReturnCode' a bakınız.
- Yanıt: "Error" (Hata)
ProcReturnCode "99" alır. bu mesajda; işlemin sanal POS sağlayan bankanın doğrulama adımıyla takıldığını gösterir. ErrMsg parametreleri hatanın detaylı açıklamasını verir.

2.3 Hash Kontrolü

Üye iş yeri parametreleri aldıktan sonra, parametreleri doğrulamak için üye iş yeri sunucusunda bir Hash kontrol edilmelidir. Hash kontrolünü sağlamak için mesaj sadece Nestpay'den gönderilir. Hashleme yapısı için birden fazla yöntem bulunuyor, bu yöntemlerden

en güncel olup mevcutta kullanılan yöntem Hash3 dediğimiz hashleme algoritmasıdır. Bu algoritmayı incelemek için Hash v3 dökümanını inceleyebilirsiniz.

3. Kod Örnekleri

3D modeli alanları için süreçleri izleyiniz. Değerler test amaçlı eklenmiştir. 3D modeli örnek kodlar içine eklenmiştir. Üye iş yerleri, hesap değişikliklerini dikkate alarak değerleri tanımlamalıdır. Bu kodlar referans olması açısından oluşturulmuştur.

3.1 ASP Kod Örneği

ASP Classic için yerleşik bir SHA-512 Karma Hesaplama kitaplığı yoktur. Bu nedenle ASP klasik kodunun, SHA 512 karma oluşturma için yerleşik bir kitaplığa sahip olan C# veya VB.Net kodunu kullanacak şekilde genişletilmesi gerekir.

```
<html>
<head>
  <title>3D</title>
  <meta http-equiv="Content-Language" content="tr">
  <meta http-equiv="Content-Type" content="text/html; charset=ISO8859-9">
  <meta http-equiv="Pragma" content="no-cache">
  <meta http-equiv="Expires" content="now">
</head>
<body>
  <!--#include file = "hex_sha1_js.asp" -->
  <%
    Dim clientId,amount,oid,okUrl,failUrl,
    rnd,taksit,islemtipi,storekey,hashstr,hash1,hash,storetype
    clientId="XXXXXXXXXX"
    amount = "9.95"
    oid = ""
    okUrl = "https://www.teststore.com/success.asp"
    failUrl = "https://www.teststore.com/fail.asp"          rnd    = now()
    storekey =  "xxxxxx"      storetype = "3d"      hashstr = clientId & oid & amount &
okUrl & failUrl &rnd &
    storekey hash = b64_sha1(hashstr)
  %>
<center>
```

```
<form method="post" action="https://<host_address>/<3dgate_path>">
  <table>
    <tr>
      <td>Credit Kart Number</td>
      <td><input type="text" name="pan" size="20"/>
    </tr>
    <tr>
      <td>CVV</td>
      <td><input type="text" name="cv2" size="4"
        value=""/></td>
    </tr>
    <tr >
      <td>Expiration Date Year</td>
      <td><input type="text" name="Ecom_Payment_Card_ExpDate_Year"
        value=""/>
      </td>
    </tr >
    <tr>
      <td>Expiration Date Month</td>
      <td><input type="text"
        name="Ecom_Payment_Card_ExpDate_Month" value=""/> </td>
    </tr>
    <tr>
      <td>Choosing Visa/MasterCard</td>
      <td>
        <select name="cardType">
          <option value="1">Visa</option>
          <option value="2">MasterCard</option>
        </select>
      </td>
    </tr>
    <tr>
      <td align="center" colspan="2">
        <input type="submit" value="Complete Payment"/>
      </td>
    </tr>
  </table>
```

```
<input type="hidden" name="clientid" value="<%=clientId%>"> <input
type="hidden" name="amount" value="<%=amount%>"> <input type="hidden" name="oid"
value="<%=oid%>">
    <input type="hidden" name="okUrl" value="<%=okUrl%>">
    <input type="hidden" name="failUrl" value="<%=failUrl%>">
    <input type="hidden" name="rnd" value="<%=rnd%>" >
    <input type="hidden" name="hash" value="<%=hash%>" >
    <input type="hidden" name="storetype" value="<%=storetype%>"> <input
type="hidden" name="lang" value="tr">
    <input type="hidden" name="currency" value="949">
</form>
</center>
</body>
</html>
```

3.2 .Net C# Kod Örneği

```
<%@ page language="C#" autoeventwireup="true" inherits="_3DModel, App_Web_fr4klrwv"
%>
<!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.0 Transitional//EN"
"http://www.w3.org/TR/xhtml1/DTD/xhtml1-transitional.dtd">
<html xmlns="http://www.w3.org/1999/xhtml" >
    <head runat="server">
        <title>3D Model</title>
    </head>
    <body>
        <%
            String clientId = "XXXXXXXXX";
            String amount = "9.95";
            String oid = "";
            String okUrl = "https://www.teststore.com/success.aspx";
            String failUrl = "https://www.teststore.com/fail.aspx";
            String rnd = DateTime.Now.ToString();
            String storekey="xxxxxx";
            String storetype = "3d";
            String hashstr = clientId + oid + amount + okUrl + failUrl + rnd + storekey;
            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);
String description = "";
String xid = "";
String lang = "";
String email = "";
String userid = "";
%>
<center>
<form method="post" action="https://<Host_Address>/<3dgate_path>">
<table>
<tr>
<td>Credit Card Number</td>
<td><input type="text" name="pan"
size="20"/>
</tr>
<tr>
<td>CVV</td>
<td><input type="text" name="cv2" size="4" value=""/></td>
</tr>
<tr>
<td>Expiration Date Year</td>
<td><input type="text"
name="Ecom_Payment_Card_ExpDate_Year" value=""/></td>
</tr>
<tr>
<td>Expiration Date Month</td>
<td><input type="text"
name="Ecom_Payment_Card_ExpDate_Monthvalue=""/></td>
</tr>
<tr>
<td>Choosing Visa Master Card</td>
<td>
<select name="cardType">
<option value="1">Visa</option>
<option value="2">MasterCard</option>
</select>
```

```

        </tr>
        <tr>
            <td align="center" colspan="2">
                <input type="submit" value="Complete
                Payment"/>
            </td>
        </tr>
    </table>
    <input type="hidden" name="clientid" value="<%=clientId%>"> <input
type="hidden" name="amount" value="<%=amount%>"> <input type="hidden" name="oid"
value="<%=oid%>">
    <input type="hidden" name="okUrl" value="<%=okUrl%>"> <input type="hidden"
name="failUrl" value="<%=failUrl%>"> <input type="hidden" name="rnd"
value="<%=rnd%>" > <input
    type="hidden" name="hash" value="<%=hash%>" > <input
    type="hidden" name="storetype" value="3d" > <input
    type="hidden" name="lang" value="tr">
    <input type="hidden" name="currency" value="949">
</form>
</center>
</body>
</html>

```

3.3 .Net - VB Kod Örneği

```

<%@ page language="VB" autoeventwireup="true"
    inherits="_3DModel, App_Web_fr4klrww"%>
<!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.0 Transitional//EN"
"http://www.w3.org/TR/xhtml1/DTD/xhtml1-transitional.dtd">
<html xmlns="http://www.w3.org/1999/xhtml">
    <head runat="server">
        <title>3D Model</title>
    </head>
    <body>
        <%
            Dim orgClientId As String = "9900000000000001"
            Dim orgOid As String = "ORDER256712jbs\j6b|"
            Dim orgAmount As String = "91.96"

```

```
Dim orgOkUrl As String = "https://www.teststore.com/success.php"
Dim orgFailUrl As String = "https://www.teststore.com/fail.php"
Dim orgTransactionType As String = "Auth"
Dim orgInstallment As String = ""
Dim orgRnd As String = DateTime.Now.ToString()
Dim orgCurrency As String = "949"
```

```
Dim clientId As String = orgClientId.Replace("\", "\\").Replace("|", "\\|")
Dim oid As String = orgOid.Replace("\", "\\").Replace("|", "\\|")
Dim amount As String = orgAmount.Replace("\", "\\").Replace("|", "\\|")
Dim okUrl As String = orgOkUrl.Replace("\", "\\").Replace("|", "\\|")
Dim failUrl As String = orgFailUrl.Replace("\", "\\").Replace("|", "\\|")
Dim transactionType As String = orgTransactionType.Replace("\", "\\").Replace("|", "\\|")
Dim installment As String = orgInstallment.Replace("\", "\\").Replace("|", "\\|")
Dim rnd As String = orgRnd.Replace("\", "\\").Replace("|", "\\|")
```

```
Dim currency As String = orgCurrency.Replace("\", "\\").Replace("|", "\\|")
Dim storeKey As String = "AB123456\|.Replace("\", "\\").Replace("|", "\\|")
```

```
Dim plainText As String = clientId + "|" + oid + "|" + amount + "|" + okUrl + "|" + failUrl
+ "|" + transactionType + "|" + installment + "|" + rnd + "||||" + currency + "|" + storeKey
```

```
Dim result As Byte()
Dim mixer As String
Dim sha As New System.Security.Cryptography.SHA512Managed()
result = sha.ComputeHash(System.Text.Encoding.ASCII.GetBytes(plainText))
Dim hashValue As String = Convert.ToBase64String(result)
```

```
Dim description As String = "";
Dim xid As String = "";
Dim lang As String = "";
Dim email As String = "";
Dim userid As String = "";
%>
```

<center>

```
<form method="post" action="https://<Host_Address>/<3dgate_path>">
  <table>
    <tr>
```

```

        <td>Credit Card Number</td>
        <td><input type="text" name="pan" size="20" />
    </tr>
    <tr>
        <td>CVV</td>
        <td><input type="text" name="cv2" size="4" value="" /></td>
    </tr>
    <tr>
        <td>Expiration Date Year</td>
        <td><input type="text" name="Ecom_Payment_Card_ExpDate_Year"
            value="" /></td>
    </tr>
    <tr>
        <td>Expiration Date Month</td>
        <td><input type="text"
            name="Ecom_Payment_Card_ExpDate_Month value="" /></td>
    </tr>
    <tr>
        <td>Choosing Visa Master Card</td>
        <td>
            <select name="cardType">
                <option value="1">Visa</option>
                <option value="2">MasterCard</option>
            </select>
        </td>
    </tr>
    <tr>
        <td align="center" colspan="2"><input type="submit"
            value="Complete Payment" /></td>
    </tr>
</table>
<input type="hidden" name="clientid" value="<%=orgClientId%>">
<input type="hidden" name="amount" value="<%=orgAmount%>">
<input type="hidden" name="oid" value="<%=orgOid%>">
<input type="hidden" name="okurl" value="<%=orgOkUrl%>">
<input type="hidden" name="failUrl" value="<%=orgFailUrl%>">
<input type="hidden" name="TranType" value="<%=orgTransactionType%>">
<input type="hidden" name="Instalment" value="<%=orgInstallment%>">
<input type="hidden" name="currency" value="<%=orgCurrency%>">

```

```
<input type="hidden" name="rnd" value="<%=orgRnd%>">
<input type="hidden" name="hash" value="<%=hash%>">
<input type="hidden" name="storetype" value="3D ">
<input type="hidden" name="lang" value="tr">
<input type="hidden" name="hashAlgorithm" value="ver2">
</form>
</center>
```

3.4 JSP Kod Örneği

```
<%@page contentType="text/html; charset=ISO-8859-9"%>
<!DOCTYPE HTML PUBLIC "-//W3C//DTD HTML 4.01
Transitional//EN" "http://www.w3.org/TR/html4/loose.dtd">
<html>
<head>
<meta http-equiv="Content-Type" content="text/html; charset=ISO-8859-9">
<title>3D Model</title>
</head>
<body>
<%
String clientId =
"XXXXXXXXX"; String amount =
"9.95"; String oid = "";
String okUrl = "https://www.teststore.com/success.jsp";
String failUrl = "https://www.teststore.com/fail.jsp";
String rnd = new java.util.Date().toString();
String storekey="xxxxxx";
String firmaadi = "xfirma";
String storetype="3d";
String hashstr = clientId + oid + amount + okUrl + failUrl + rnd +
storekey;
java.security.MessageDigest sha1
java.security.MessageDigest.getInstance("SHA-1");
String hash =
(newsun.misc.BASE64Encoder()).encode(sha1.digest(hashstr.getBytes()));
String description =
""; String xid = "";
String lang="";
```

```

String email="";
String userid="";
%>
<center >
<form method="post" action="https://<host_address>/<3dgate_path>">
  <table>
    <tr>
      <td>Credit Card Number</td >
      <td><input type="text" name="pan"
        size="20"/>
    </tr>
    <tr>
      <td>CVV</td>
      <td><input type="text" name="cv2" size="4" value=""/></td>
    </tr>
    <tr>
      <td>Expiration Date Year</td>
      <td><input type="text"
        name="Ecom_Payment_Card_ExpDate_Year" value=""/></td>
    </tr>
    <tr>
      <td>Expiration Date Month</td>
      <td><input type="text"
        name="Ecom_Payment_Card_ExpDate_Month" value=""/></td>
    < /tr>
    <tr>
      <td>Choosing Visa / Master Card</td>
      <td>
        <select name="cardType">
          <option value="1">Visa</option>
          <option value="2">MasterCard</option>
        </select>
      < /tr >
    <tr>
      <td align="center" colspan="2">
        <input type="submit" value="Complete Payment"/>
      <
    </td >
  </tr>

```

```
</table>
<input type="hidden" name="clientid" value="<%=clientId%>"> <input
type="hidden" name="amount" value="<%=amount%>">
<input type="hidden" name="oid" value="<%=oid%>">
<input type="hidden" name="okUrl" value="<%=okUrl%>">
<input type="hidden" name="failUrl" value="<%=failUrl%>"> <input type="hidden"
name="rnd" value="<%=rnd%>" > <input type="hidden" name="hash"
value="<%=hash%>" <input
type="hidden" name="storetype" value="3d" > <input
type="hidden" name="lang" value="tr">
<input type="hidden" name="currency" value="949">
</form>
</center>
</body>
</html>
```

3.5 PHP Kod Örneği

```
<html>
<head>
<title>3D</title>
<meta http-equiv="Content-Language" content="tr">
<meta http-equiv="Content-Type" content="text/html; charset=ISO-8859-9">
<meta http-equiv="Pragma" content="no-cache">
<meta http-equiv="Expires" content="now">
< /head >
<body>
<?php
$clientId = "XXXXXXXXX";
$amount = "9.95";
$oid = "";
$okUrl = "https://www.teststore.com/success.php";
$failUrl = "https://www.teststore.com/fail.php";
$rnd = microtime();
$storekey = "xxxxxx";
$storetype = "3d";
$hashstr = $clientId . $oid . $amount . $okUrl . $failUrl . $rnd . $storekey;
```

```
$hash = base64_encode(pack('H*',sha1($hashstr)));
$description =
""; $xid = "";
$lang="";
$email="";
$userid="";
?>
<center >
<form method="post"
action="https://<host_address>/<3dgate_path>">
<table>
  <tr>
    <td>Credit CardNumber</td>
    <td><input type="text" name="pan" size="20"/>
  </tr>
  <tr>
    <td>CVV</td>
    <td><input type="text" name="cv2" size="4" value=""/></td>
  </tr>
  <tr>
    <td>Expiration Date Year</td>
    <td><input type="text"
      name="Ecom_Payment_Card_ExpDate_Year" value=""/></td>
  </tr>
  <tr>
    <td>Expiration Date Month</td>
    <td><input type="text"
      name="Ecom_Payment_Card_ExpDate_Month" value=""/></td>
  </tr>
  <tr>
    <td>Choosing Visa / Master Card</td>
    <td>
      <select name="cardType">
        <option value="1">Visa</option>
        <option value="2">MasterCard</option>
      </select>
    </td>
  </tr>
  <tr>
```



```
<td align="center" colspan="2">
  <input type="submit" value="Complete
    Payment"/>
</td>
</tr>
</table>
<input type="hidden" name="clientid" value="<?php echo $clientId ?>"> <input
type="hidden" name="amount" value="<?php echo $amount ?>"> <input type="hidden"
name="oid" value="<?php echo $oid ?>">
  <input type="hidden" name="okUrl" value="<?php echo $okUrl ?>"> <input
type="hidden" name="failUrl" value="<?php echo $failUrl ?>"> <input type="hidden"
name="rnd" value="<?php echo $rnd ?>" >
  <input type="hidden" name="hash" value="<?php echo $hash ?>" > <input
type="hidden" name="storetype" value="3d" >
  <input type="hidden" name="lang" value="tr">
  <input type="hidden" name="currency" value="949">
</form>
</center>
</body>
</html>
```

APPENDIX A: Geçit Parametreleri

Zorunlu Parametreler

Alan	Açıklama	Format
clientid	Üye iş yeri numarası	Harf ya da rakam, maksimum 15 karakter
storetype	Üye iş yerinin ödeme modeli	
islem tipi	İşlem Tipi	Harf ya da rakam, geçerli değerler{Auth, PreAuth, PostAuth, Void, Credit}

amount	İşlem tutarı	Rakam, ondalık rakamlar ",", veya "." ile ayrılır. Karakter kullanılmaz.
currency	ISO para birimi kodu	Rakam, 3 rakam (TL için 949)
oid	Sipariş numarası	Harf ya da rakam, maksimum 64 karakter
pan	Kredi Kartı Numarası	Maksimum 20 rakam
Ecom_Payment_Card_ExpDate_Year	Son kullanma tarihi (yıl)	4 rakam
Ecom_Payment_Card_ExpDate_Month	Son kullanma tarihi (ay)	2 rakam
okUrl	Nestpay Ödeme Geçidine gelen başarılı işlem bilgilendirmesini üye iş yeri tarafındaki başarılı işlem bildirimi içinönceden belirlenmiş olan URL' e iletilir.	Örnek: http://www.test.com/basarili.php
failUrl	Nestpay Ödeme Geçidine gelen başarısız işlem bilgilendirmesini üye iş yeri tarafındaki başarısız işlem bildirimi için belirlenmiş URL'e iletir.	Örnek: http://www.test.com/basarisiz.php
lang	Nestpay ödeme sayfasında kullanılan dil.	Türkçe için "tr" , İngilizce için "en"
rnd	Hash karşılaştırması için kullanılacak rastgele dizedir	Sabit uzunluk, 20 karakter
hash	Kullanıcı doğrulama için Hash değeri	

İsteğe Bağlı Parametreler

Parameter	Description	Format
refreshtime	Saniye cinsinden yönlendirme sayacı değeri (okUrl veya failUrl'ye)	

Encoding	Gönderilen verinin kodlaması. Default değeri utf-8dir	En Fazla 32 karakter
description	açıklama	En Fazla 255 karakter
instalment	Taksit adedi	Sayı
email	Müşteri eposta adresi	En Fazla 64 karakter
firmaadi	Faturalama yapılacak firma adı	Maksimum 255 karakter
Faturafirma	Faturalama yapılacak müşteri adı	Maksimum 255 karakter
tel	Faturalama yapılacak telefon numarası	Maksimum 32 karakter
Fadres	Fatura 1. adres satırı	Maksimum 255 karakter
Fadres2	Fatura 2. adres satırı	Maksimum 255 karakter
Filce	Faturalama yapılacak ilçe	Maksimum 64 karakter
Fil	Faturalama yapılacak şehir	Maksimum 32 karakter
Fpostakodu	Faturalama yapılacak posta kodu	Maksimum 32 karakter
Fulke kodu	Faturalama yapılacak ülke kodu	Maksimum 3 karakter
NakliyeFirma	Teslimat yapılacak firma adı	Maksimum 255 karakter
tismi	Teslimat yapılacak müşteri adı	Maksimum 255 karakter
tadres	Teslimat 1. adres satırı	Maksimum 255 karakter
tadres2	Teslimat 2. adres satırı	Maksimum 255 karakter
tilce	Teslimat yapılacak ilçe	Maksimum 64 karakter
til	Teslimat yapılacak şehir	Maksimum 32 karakter
tpostakodu	Teslimat posta kodu	Maksimum 32 karakter
tulke kod	Teslimat yapılacak ülke kodu	Maksimum 3 karakter
id1	1. ürün Id	Maksimum 128 karakter
itemnumber1	1. ürün numarası	Maksimum 128 karakter
productcode1	1. ürün üretim kodu	Maksimum 64 karakter
qty1	1. ürün miktarı	Maksimum 32 karakter
desc1	1. ürünün açıklaması	Maksimum 128 karakter
price1	1. ürün birim fiyatı	Maksimum 32 karakter
amount1	1. ürün miktarı X 1. ürün birim fiyatı	Maksimum 32 karakter

İşlem Yanıt Parametreleri

Parameter	Description	Format
-----------	-------------	--------

AuthCode	İşlem doğrulama / Onaylama kodu	6 karakter
Response	Ödeme durumu	Olası değerler: "Approved", "Error", "Declined"
HostRefNum	Banka referans kodu	12 karakter
ProcReturnCode	İşlem durum kodu	2 basamaklı, onaylanmış: "00" Nestpay hatası: "99" Diğer hatalar: ISO-8583 hata kodları
TransId	İşlem numarası	Maksimum 64 karakter
ErrMsg	Hata mesajı	Maksimum 255 karakter
ClientIp	Müşterinin IP adresi	Maksimum 15 karakter
ReturnOid	Dönen sipariş numarasıdır ve girilen ile aynı olmalıdır	Maksimum 64 karakter
MaskedPan	Maskelenmiş kredi kartı numarası	12 karakter, örnek: XXXXXX***XXX
EXTRA.TRXDATE	İşlem Tarihi	17 karakter, "yyyyMMdd HH:mm:ss"
rnd	Hash karşılaştırması için kullanılacak rastgele dizedir	Sabit uzunluk, 20 karakter
HASHPARAMS	Hash hesaplaması için kullanılan alan adlarını içerir. Alan adları ":" karakteri ile eklenir.	
HASHPARAMSVAL	Hash hesaplaması için eklenen hash alan değerlerini içerir. Alan değerleri HASHPARAMS alanında aynı sırayla eklenir.	
HASH	HASHPARAMSVAL ve müşteri şifre alanları için hash	

MPI Yanıt Parametreleri

Alan	Açıklama	Format
mdStatus	3D işlem durum kodu	1 = Doğrulanmış İşlem (Full 3D) 2-3-4: kart kayıtlı değil (half3d) 5,6,7,9: Geçerli doğrulama veya

		sistem hatası 0: doğrulama başarısız
merchantID	Üye iş yeri numarası (MPI)	15 karakter
txstatus	3D status for archival	olası değerler "A", "N", "Y"
iReqCode	İsteği doğrulamak için verinin düzgün formatlandığını gösteren ACS tarafından sağlanan kod	2 basamaklı, rakam
iReqDetail	Geçersiz İstek Kodunda bazı veri elemanlarını tanımlayan detay verisi	
vendorCode	iReqDetail hatasını tanımlayan hata mesajı	
PAREsSyntaxOK	Eğer PAREs doğrulaması sentaktik olarak doğruysa bu değer doğrudur, aksi takdirde yanlıştır	"Y" veya "N"
ParesVerified	Eğer imza doğrulamasının sonucu başarılıysa bu değer doğrudur. Eğer PAREs mesajı alınmamışsa ya da imza doğrulaması başarısız olursa yanlıştır	"Y" veya "N"
eci	Elektronik Ticaret Göstergesi (Electronic Commerce Indicator)	2 basamaklı, 3D siz işlemlerde boş
cavv	Kart güvenlik numarası (CardholderAuthentication Verification Value, ACS tarafından belirlenir.)	28 karakter, Base64 ile kodlanmış bir 20 byte değerini içeren 28 byte sonuç verir)
xid	Benzersiz internet işlem numarası	28 karakter, base64 kodlaması
cavvAlgorithm	Kart güvenlik numarası algoritması	olası değerler "0", "1", "2", "3"
md	Kart numarası yerine MPI verisi	Alpha-nümerik
Version	MPI versiyon bilgisi	3 karakter (örn: "2.0")
sID	Şema numarası	Visa için "1", Mastercard için "2"
MdErrorMsg	Eğer var ise MPI dan gelen hata mesajı	Maksimum 512 karakter