


BonusFlaş – GarantiPay Uygulama Detay ve Teknik Entegrasyon Dokümanı:

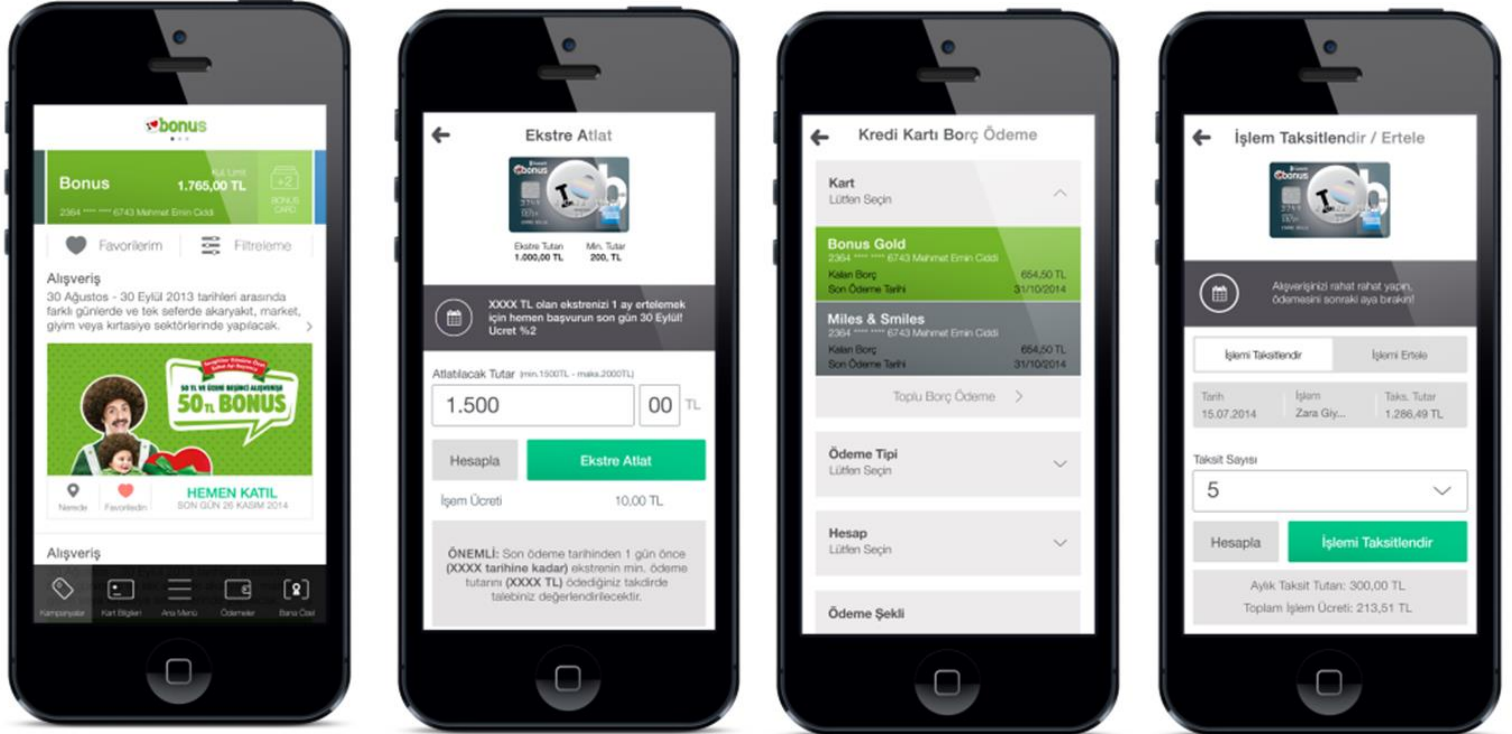
BonusFlaş Nedir?

- BonusFlaş, kredi kartı müşterilerimizin kartlarıyla ilgili tüm işlemlerini tek bir kanaldan yapmalarına olanak sağlayan bir mobil uygulamadır.
- Müşterilerimiz, kampanyalarımızı görüntüleyebilecek, katılabilecek ve kampanyadaki durumlarını takip edebilecekler.
- Kart limiti ve güncel işlemler gibi bilgileri takip edip, işlem taksitlendire, ekstre atlat gibi servislerimizden faydalanabilecekler.
- Kart borçlarını ödeyebilecekler ve internetten alışveriş yapabilecekler.
- Kişiyi özel ve/veya lokasyona özel çıkacağımız kampanyalardan ve tekliflerden faydalanabilecekler.

BonusFlaş' ın Müşterilerimiz ve Üye İşyerlerimize Faydaları:

- Müşterilerimiz bu platform üzerinden kartlarına ait tüm işlemleri kolayca takip edebilecekler. (Limit Görüntüleme, Limit Arttırım, Kampanyalara Katılım, Borç Ödeme vb.)
- Müşterilerimizin harcama alışkanlıklarına göre ilgili kampanyaların gösterim sıklığının otomatik olarak yönetilmesine imkan veren Akıllık Kampanya Gösterim modülü.
- Müşterilerimizin seçimleri ve harcama alışkanlıklarına göre özelleştirilmiş Bana Özel Kampanyalar menüsü.
- Genel ve Sektör kampanyalarımıza ek olarak üye işyerlerimize özel tüm kampanyalara da uygulama üzerinden erişim ve katılım mümkün olacak.
- Müşterimizin lokasyonuna göre bu lokasyonda yer alan üye işyerlerimizin bilgileri ve kampanya detayları, müşterilerimize eş zamanlı olarak uygulama üzerinden gösterilecek.
- Müşterilerimizin ödeme alışkanlıkları ve kampanya tercihleri doğrultusunda oluşturulan yapı, üye işyerlerimizin doğrudan hedef kitlelerine ulaşmalarını sağlayacak kampanyaların oluşturulabilmesine olanak sağlayacak. Kampanya bildirimlerinin hedef kitledeki kullanıcılara ulaşması da sağlanmış olacak.
- Müşterilerimiz Uygulama Üzerindeki **GarantiPay*** Cüzdan özelliği sayesinde üye işyerlerimizin Web siteleri ve mobil uygulamalarında kart bilgilerini paylaşmadan kolayca kart seçerek ödemelerini tamamlayabilecekler.

*  ile ilgili kapsamlı bilgileri 2.sayfada bulabilirsiniz.



Görsel 1 – Müşterilerimize Özel Kampanya Öneri ve Kart İşlem Ekranları



Nedir?

- GarantiPay, sanal alışveriş dünyasında yeni bir ödeme deneyimi yaratacak olan bir ödeme aracıdır.
- Sanal alışveriş sırasında, ödeme seçenekleri adımına eklenecek bir check out olarak yer alır.
- Hem e-commerce(in-web) işlemlerde, hem de m-commerce(in-app) işlemlerde müşteriler bu deneyimi yaşayabilir.
- GarantiPay check out seçildiğinde, müşteri BonusFlaş'e yönlendirilir, işlemini BonusFlaş içinden sonlandırır.

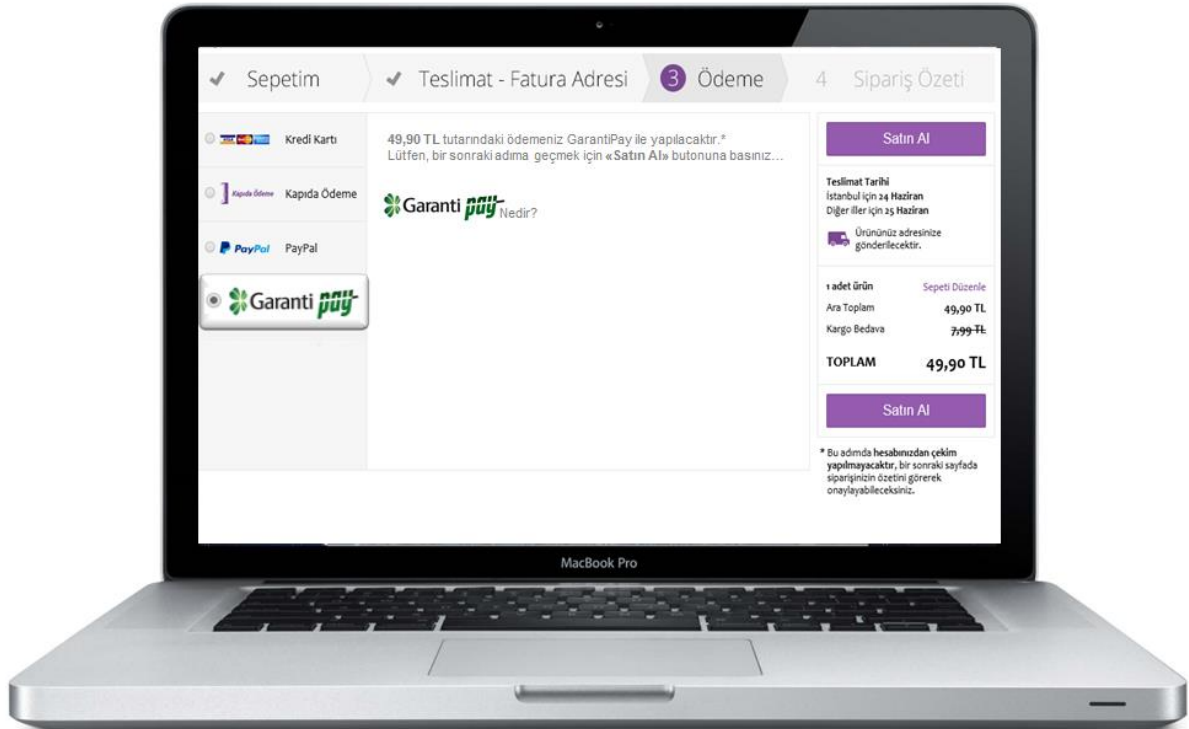


' in Müşterilerimiz ve Üye İşyerlerimize Faydaları:

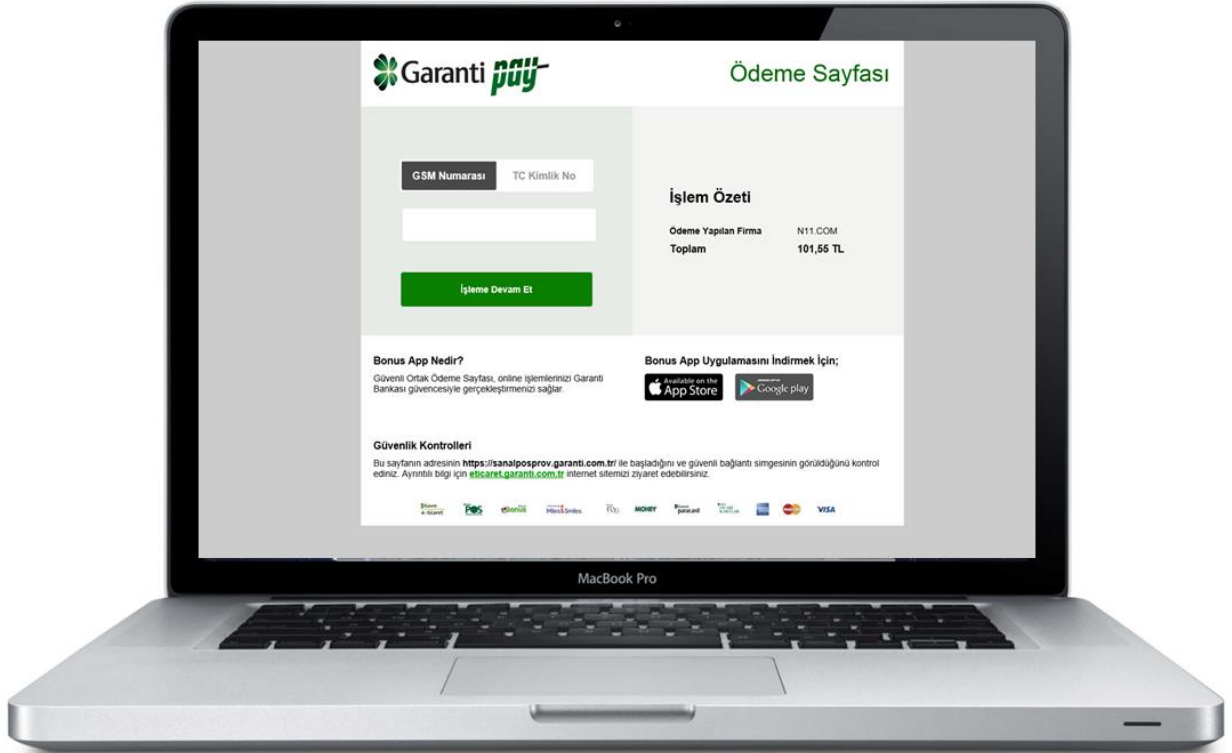
- Müşterilerimize kolay, hızlı ve güvenli bir ödeme deneyimi sağlar.
- Müşterilerimizin işlem esnasında cüzdanlarını ve kartlarını yanlarında bulundurma ihtiyacını ortadan kaldırır.
- Ödeme işlemi esnasında kart bilgilerinin üye işyerleri ile paylaşılması ihtiyacı ortadan kaldırır.
- OneClickPayment yapısını kullanmak isteyen ama kart bilgilerini saklamak istemeyen üye işyerlerimiz için bu yapıyı hem mobil uygulamalarından hem de web sitelerinden yapılacak ödemeler için sağlar.
- Mobil Uygulama üzerinden de ürün ve servislerini sağlayan üye işyerlerimize de ödeme çözümü sağlar. Mobil Uygulama üzerinden kredi kartı bilgisinin manuel girişi ile ödemenin yapıldığı yapıya göre işlem sürelerini %70 oranında kısaltarak kullanıcı deneyimine olumlu katkı sağlar.

Web Sitesi Üzerinden GarantiPay ile Ödeme(InWeb):

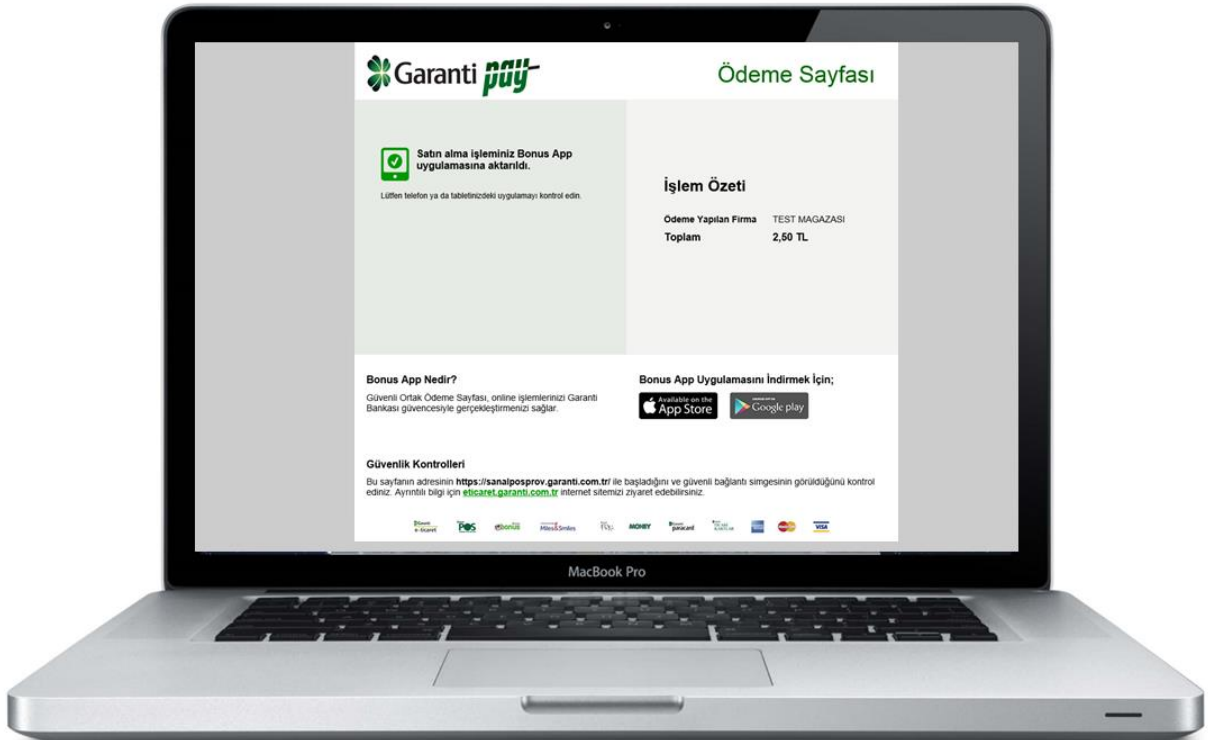
1. Müşterimiz Sepetini oluşturur ve ödeme seçenekleri adımından GarantiPay' i seçer.



2. GarantiPay Ödeme Sayfası Açılır ve Müşterimiz GSM numarasını ekrana girerek işleme devam eder.



3. Müşterimiz, işlemin BonusFlaş uygulamasına aktarıldığı, mobil cihazından uygulamayı kontrol etmesi gerektiği ile ilgili mesajı görür.



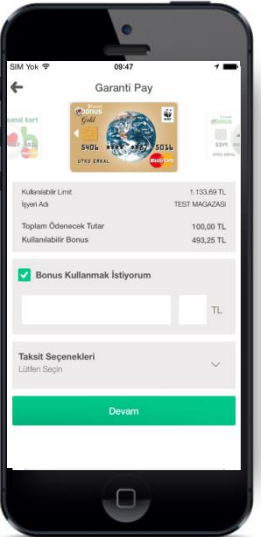
(MOBİL CİHAZ ÜZERİNDE BİLDİRİMLER AÇIKSA...)



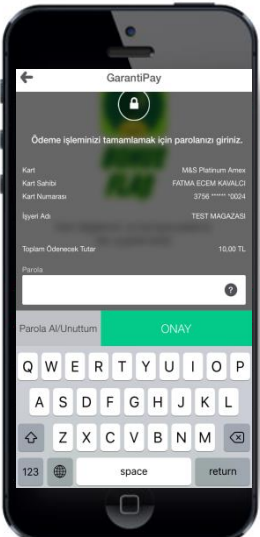
4. Müşterimizin, uygulamasının kurulu olduğu mobil cihazına uyarı ulaşır. Bu aşamada GarantiPay Uygulaması iki durumda olabilir:

Soft Login: Uygulamaya Kimlik Doğrulaması Yapılarak Giriş yapılmış, son işlem üzerinden 8 dakikalık süre geçmiş olduğundan işlemlere devam edebilmek müşterilerimizin parola girmesi gerekmektedir.

Hard Login: Uygulamaya Kimlik Doğrulaması Yapılarak Giriş yapılmış, son işlem üzerinden 8 dakikalık süre henüz dolmamıştır.



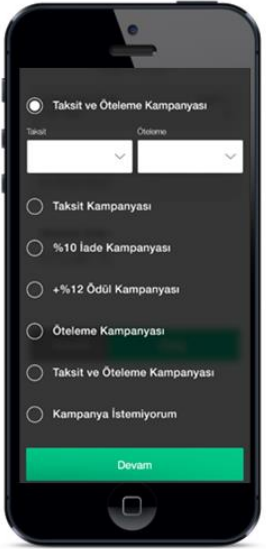
5.Uyarı Üzerinden Uygulamaya giriş yaptıktan sonra Kart Seçim Ekranından, ödeme yapmak istediği kartı seçer. Bonus kullanmak istiyorsa kullanmak istediği Bonus tutarını girer, işlemi taksitli yapmak istiyorsa istediği taksit adedini seçerek devam eder.



6.Bu ekran sadece, Müşterimiz Soft Login durumunda ise görüntülenir. Parola girişi sonrasında işlem doğrudan provizyona yönlendirilir. Hard Login durumunda ise, bu ekran açılmadan doğrudan bir sonraki ekran görüntülenir.



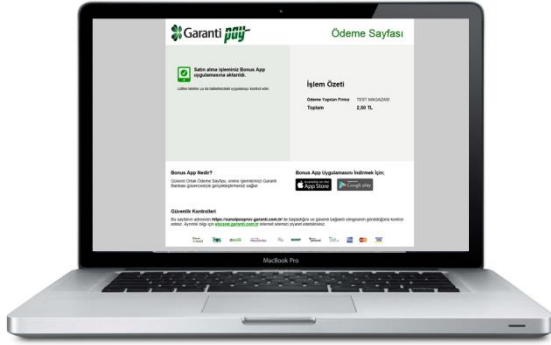
7. Hard Login Durumundaki Müşterimiz doğrudan bu işlem detayları ekranında Ödeme Yap' ı seçerek devam eder. İşlem Provizyona yönlendirilir.



8. Provizyon İşlemi sonrasında müşterimize sunulabilecek bir kampanya varsa bunlar gösterilir. Müşterilerimiz isterlerse kampanya seçimi yapabilirler.

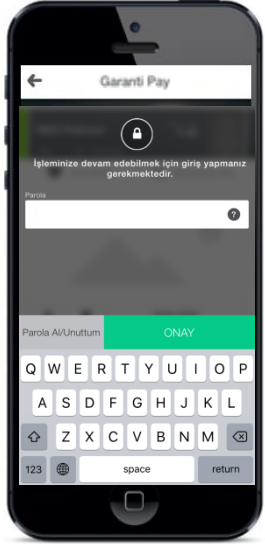


9. Sonrasında ödeme işlemi tamamlanır ve müşterimiz web sitesinde işlemin başarı ile tamamlandığı mesajını görür.

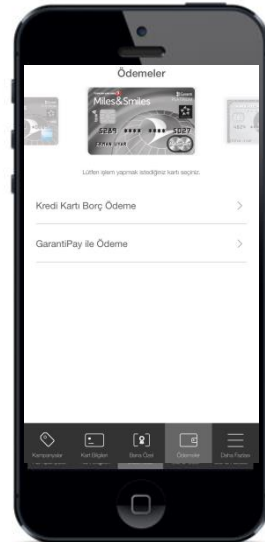


(MOBİL CİHAZ ÜZERİNDE BİLDİRİMLER KAPALIYSA veya BİLDİRİM ULAŞMAZSA...)

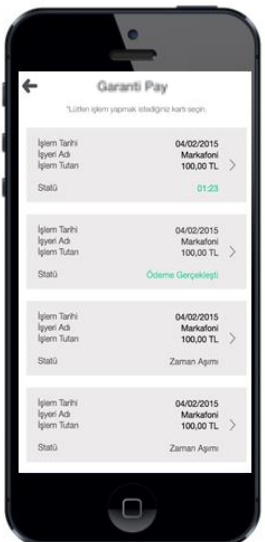
3. Satın Alma İşleminin mobil uygulamaya yönlendirildiğine dair Web sitesinde verilen bilgiyi takiben, müşterimiz mobil cihazından GarantiPay Uygulamasını açar,



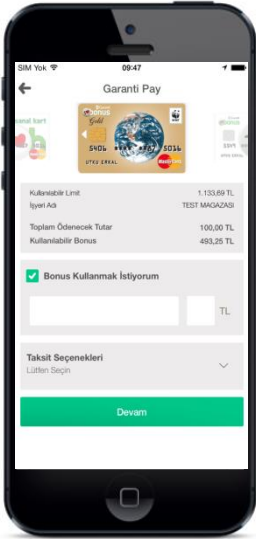
4. Satın Alma İşleminin mobil uygulamaya yönlendirildiğine dair Web sitesinde verilen bilgiyi takiben, müşterimiz mobil cihazından GarantiPay Uygulamasını açar, eğer soft login modunda ise yandaki ekran açılır, parola girer. Hard Login durumunda ise doğrudan bir sonraki ekran görüntülenir.



5. Açılan ekranın en altında bulunan menünün ortasında yer alan "Ödemeler" sekmesine giriş yaparak, Müşterimiz GarantiPay ile Ödeme' yi seçer.



6. Açılan Ekranda GarantiPay üzerindeki Ödemeler arasından ödeme yapmak istediği işlemine ait olanı seçer.



7. Açılan, Kart Seçim Ekranından, ödeme yapmak istediği kartı seçer. Bonus kullanmak istiyorsa kullanmak istediği Bonus tutarını girer, işlemi taksitli yapmak istiyorsa istediği taksit adedini seçerek devam eder.

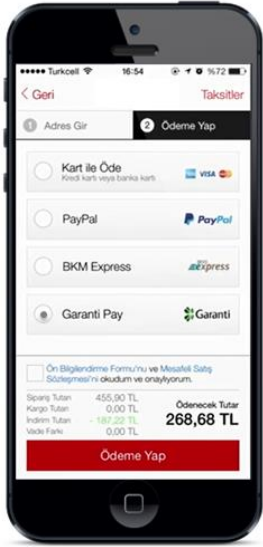


8. Provizyon işlemi sonrasında müşterimize sunulabilecek bir kampanya varsa bunlar gösterilir. Müşterilerimiz isterseler kampanya seçimi yapabilirler.

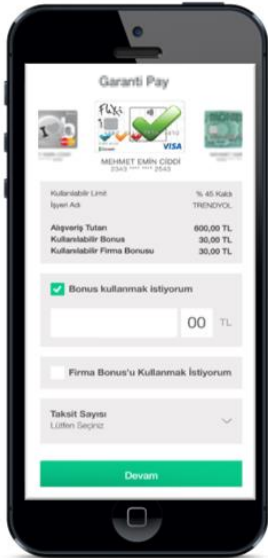


9. Sonrasında ödeme işlemi tamamlanır ve müşterimiz web sitesinde işlemin başarı ile tamamlandığı mesajını görür.

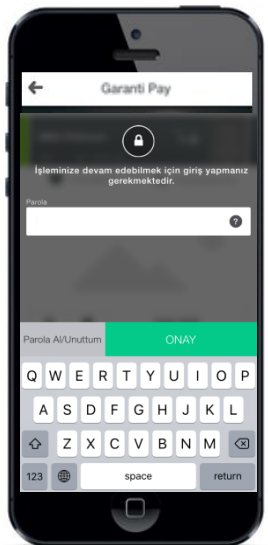
Mobil Uygulama Üzerinden GarantiPay ile Ödeme(InAPP):



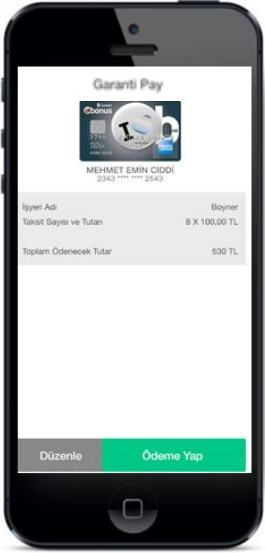
1. Müşterimiz Üye İşyerimizin Mobil Uygulamasında siparişini oluşturduktan sonra ödeme alternatifleri arasından GarantiPay' i seçer.



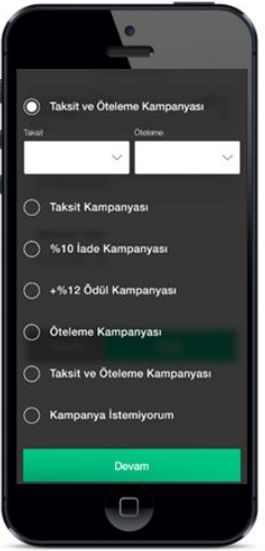
2. Kart Seçim Ekranından, ödeme yapmak istediği kartı seçer. Bonus kullanmak istiyorsa kullanmak istediği Bonus tutarını girer, işlemi taksitli yapmak istiyorsa istediği taksit adedini seçerek devam eder.



3. Bu ekran sadece, Müşterimiz Soft Login durumunda ise görüntülenir. Parola girişi sonrasında işlem doğrudan provizyona yönlendirilir. Hard Login durumunda ise, bu ekran açılmadan doğrudan bir sonraki ekran görüntülenir.



4. Hard Login Durumundaki Müşterimiz doğrudan bu işlem detayları ekranında Ödeme Yap' ı seçerek devam eder. İşlem Provizyona yönlendirilir.



5.Provizyon İşlemi sonrasında müşterimize sunulabilecek bir kampanya varsa bunlar gösterilir. Müşterilerimiz isterseler kampanya seçimi yapabilirler.



6. Sonrasında ödeme işlemi tamamlanır ve müşterimiz alışveriş yaptığı uygulamaya yönlendirilir.

UYGULAMAYI İNDİRMEK İÇİN:

Android veya IOS için test uygulamasını indirmek için uygulamanın kullanılacağı IP bilgisinin ve uygulamayı indirme talebiniz ile beraber eticaret@garanti.com.tr veya garantieticaret@garanti.com.tr adresine iletilmesi gerekmektedir.

Gelecek olan teyidi takiben aşağıdaki link üzerinden uygulamayı indirerek test etmeye başlayabilirsiniz:

IOS: (Açılan sayfada à Download BonusApp)

<https://bonusgw-tst.garanti.com.tr/bonusapp/downloadBonusApp.json>

Android:

Bu URL'den indireceğiniz .apk uzantılı dosya telefonun ana menüsündeki "Downloads" içerisine inmiş olacak. Ona tıklayıp uygulamayı yüklemeniz gerekiyor

Açılan sayfada à Download BonusAppAndroid

<https://bonusgw-tst.garanti.com.tr/bonusapp/downloadBonusAppAnd.json>

*iOS'tan farklı olarak, bu URL'den indireceğiniz .apk uzantılı dosya telefonun ana menüsündeki "Downloads" içerisine indiriliyor, buradan kurulum yapmanız gerekiyor.

Test Uygulaması için Kullanıcı Bilgileri:

Müşteri Numarası: 33838408 / **TCKN:** 13775068052

Parola: garanti1

GSM: 5358862329

Test kullanıcısı ile ilgili destek talebiniz olması halinde eticaret@garanti.com.tr adresi üzerinden bilgi temin edebilirsiniz.

GARANTIPAY ENTEGRASYONU TEKNİK DETAYLAR:

Web Üzerinden GarantiPay İsteği oluşturulması

Web sayfaları üzerinden GarantiPay isteklerinin başlatılması için 3D secure ve OOS akışlarında kullanılan html post yapısı kullanılacaktır.

Bu yapıda istekler;

Test ortamı için : <https://sanalposprovtest.garanti.com.tr/servlet/gt3dengine>

Prod ortamı için : <https://sanalposprov.garanti.com.tr/servlet/gt3dengine>

Adreslerine yollanılacaktır.

Test ortamı için mağaza bilgileri

İşyeri numarası : 3424113

Terminal numarası : 30690133

Provoos kullanıcısı için şifre : 123qweASD/

3d secure key : 12345678

Şeklinde olacaktır.adresine html post metodu ile yollanmaktadır. 3D secure ve OOS akış detayları için <http://www.garantipos.com.tr/mailling/GVP.zip> adresinden indirilecek dokümanlar dikkate alınabilir.

GarantiPay işlemlerinde

secure3dsecuritylevel alanında “CUSTOM_PAY” ifadesi yollanılacaktır.

Txntype alanında “gpdatarequest” ifadesinin kullanılması gerekecektir.

txnsubtype alanında provizyon işleminde kullanılacak işlem tipi bulunacaktır. Örneğin satış işlemi için bu alanda “sales” ifadesinin gönderilmesi sağlanacaktır.

Companyname alanında App üzerinde gösterilecek “işyeri adı” yollanılacaktır.

Garantipay alanında “Y” ifadesi yollanılacaktır.

İşyeri gönderilen işlemde Bonus ve Firma bonus kullanımlarına izin verilip verilmeyeceğini kendisi belirleyecektir. Aşağıda verilen 2 alan için “Y” değerinin gönderilmesi işlemde ödül kullanımına izin verildiği anlamına gelecektir. İşyerinin banka üzerinde yetkileri tanımlı ise APP üzerinden kullanıcıya ödül kullandırımı için alanlar açılacaktır.

bnsuseflag ve fbbuseflag alanları işyeri tarafından doldurulacaktır.

GarantiPay işlemlerinde işyeri tarafında işlemde kullanılabilecek taksitler tutarları ile birlikte Sanalpos üzerinden iletilecektir. App üzerinde işyerinin gönderdiği taksit sayıları ve tutarları gösterilerek müşterinin bu işlemler için seçim yapması sağlanacaktır.

Txnamount alanında peşin işlem tutarı son 2 hanesi kuruş olacak şekilde kuruş ayrımı olmadan yollanılacaktır.

TxnTimeOutPeriod alanında app üzerinde GarantiPay akışlarında işlemin ne kadarlık bir süre içerisinde onaylanabilir olarak kalacağı işyeri tarafından yollanılacaktır. Bu alanda gelen süre saniye cinsinden olacaktır. İşyeri üzerinden , sanalPoS’a, SanalPoS tarafından Host tarafına bu bilgi taşınacaktır. Gelen süre ile host tarafında işlem detayında tutulan expiredate alanındaki değerin hesaplanması sağlanacaktır. Bu süre, işlemin geliş zamanına , istek içerisinde gelen **TxnTimeOutPeriod** değerinin eklenmesi ile bulunacaktır.

XML mesajlaşmasında gönderilen taksit bilgilerinin sadece ticari kartlar için gösteriminin yapılması gerektiğini belirtmek için “InstallmentOnlyForCommercialCard” alanı konulacaktır. Bu alanda Y değerinin gelmesi halinde, gönderilen taksit değerlerinin App üzerinde sadece ticari kart seçildiğinde geçerli olacağı anlamına gelecektir

İşlem içerisinde kaç farklı taksit seçimi olacağı **totallinstallmentcount** alanında yollanılacaktır. Örneğin burada 3 değeri geliyorsa işlem içerisinde 3 farklı taksit değerinin olduğu anlaşılacaktır. Buna göre aşağıda verilen değerlerin okunması sağlanacaktır. Totallinstallmentcount alanında yollanılan taksit değerlerinden düşük bir değer gelirse, totallinstallmentcount alanında gönderilen sayı kadar değer okuması yapılacaktır. Yollanılan diğer taksit değerleri dikkate alınmayacaktır.

Taksit numarası **installmentnumberX** alanında yollanılacaktır. **installmentamountX** alanında da ilgili taksit tutarı bulunacaktır.

installmentratewithreward alanında eğer işyeri Bonus kullanımlarında taksit tutarını yeniden hesaplıyor ise bu hesaplamada kullanılan taksit tutarı hesaplama oranı yollanılacaktır. Bu alan dolu gelirse App üzerinde peşin işlem

kullanılan ödül miktarı düşülecek ve gönderilen oran ile kalan tutar kullanılarak yeni taksitli işlem tutarı hesaplanacaktır. Taksit gösterimlerinde ve provizyona işlemin gönderiminde yeni tutarın kullanılması sağlanacaktır.

Her bir taksit için bu değerlerin yollanılması sağlanmalıdır. Örneği 2, 4 ve 6 taksit seçimi sunan bir işyeri için

Txnamount:10000

totalinstallmentcount = 3

installmentnumber1=2

installmentamount1 = 10200 //102 TL için 10200 olarak girilmelidir. Bu örnekte 102.00 TL için 10200

installmentratewithreward1=200 // %2 için 200 girilmelidir. %X.XX Formatında çalışmaktadır. Bu Örnekte %2.00 için 200

installmentnumber2=4

installmentamount2 = 10400

installmentratewithreward2=400

installmentnumber3=6

installmentamount3 = 10600

installmentratewithreward3=600

şeklinde yollanılacaktır.

App üzerinden

Peşin 100,00 TL

2 taksit : 102,00TL

4 taksit : 104,00 TL

6 taksit : 106,00 TL

Şeklinde gösterim yapılacak ve gösterilen değerler üzerinden müşterinin yaptığı seçim ile provizyon alınacaktır.

Provizyon alındıktan sonra provizyon cevabı işyeri tarafından iletilen Success ve Error URL adreslerine döndürülecektir.

Dönen istek içerisinde BonusFlaş'a özel olarak eklenen

- gpinstalment
- gpisntallmentamount
- usedbonusamount
- usedfbbamount
- **cardref**

alanları dikkate alınmalıdır. Bu alanda müşterinin

- seçtiği taksit sayısı
- taksit tutarı
- Bonus kullanımı yaptıysa kullandığı bonus tutarı
- Firma bonus kullanımı yaptıysa kullanılan firma bonus tutarı
- Banka tarafından dönülecek tekil kart bilgisidir.

yer alacaktır.

Cevap mesajında bu bilgileri ile farklı bir hash değeri dönülecek ve işyerinin bu değeri doğrulaması hack saldırılarına karşı işyerini koruyacaktır .

Hash doğrulaması :

Success veya Error URL ye dönen isteklerde GarantiPaY isteklerinde standart **hashdata** değerine ek olarak "garantipayhashdata" değeri gönderilecektir. Bu değer içerisinde müşterinin seçtiği taksit ve taksite ait tutar değeri ile hesaplanan hash değeri yollanılacaktır.

İşyerinden garantPaY işlemlerinde bu değeri doğrulaması istenecektir .

garantipayhashdata değeri için kullanılacak gphashparamsval değerleri aşağıdaki değerlerinin yan yana getirilmesi ile elde edilecektir.

clientId + oid + authcode + procreturncode + gpinstallmentamount + gpinstallment + Storekey i hash edip çıkan sonucu, işlem sonucu dönen gphashdata ile karşılaştırıyoruz,
Eşit ise bankadan geliyor.
procreturncode 00 değilse gphashdata gelmez, başarısız bir işlemi kontrol etmeye gerek olmayacaktır.

Program akışı doğrulama

```
string strType = Request.Form.Get("txntype");
if (strType == "gpdatarequest")
{
    string okmu = "";
    string hvl = "";
    Response.Write("<br></br>");
    Response.Write("İşleminiz BonusPay base64 değerleri: ");
    Response.Write("clientId + oid + authcode + procreturncode + gpinstallmentamount + gpinstallment+ Storekey");
    Response.Write("<br></br>");
    hvl = strTerminalID + strOrderID + Request.Form.Get("authcode") + Request.Form.Get("procreturncode") + Request.Form.Get("gpinstallmentamount") + Request.Form.Get("gpinstallment") + anahtar;
    Response.Write(hvl);
    Response.Write("<br></br>");
    string param1 = hvl;
    System.Security.Cryptography.SHA1 sha1 = new System.Security.Cryptography.SHA1CryptoServiceProvider();
    byte[] hashbytes1 = System.Text.Encoding.GetEncoding("ISO-8859-9").GetBytes(param1);
    string s = System.Text.Encoding.UTF8.GetString(hashbytes1, 0, hashbytes1.Length);
    Response.Write("<br></br>");
    Response.Write(s);
    byte[] inputbytes1 = sha1.ComputeHash(hashbytes1);
    String hashCalculated1 = Convert.ToBase64String(inputbytes1);
    Response.Write("<br></br>");
    Response.Write("Formüle göre sonuç: ");
    Response.Write(hashCalculated1);
    Response.Write("<br></br>");
    okmu = Request.Form.Get("gphashdata");
    Response.Write("Bankadan dönen gphashdata: ");
    Response.Write(okmu);
    if (hashCalculated1.Equals(okmu))
    {
        Response.Write("<br></br>");
        Response.Write("!!!!BONUS PAY KONTROL BANKADAN GELİYOR!!!!");
        Response.Write("<br></br>");

        if (Request.Form.Get("procreturncode") == "00")
        {
```

```
//Mesaj bankadan geliyor ve onaylı akışı
```

```

    }

    Response.Write("<br></br>");
}
else
{
    if (Request.Form.Get("procreturncode") == "00")
    {
        Response.Write("<br></br>");
        Response.Write("!!!!!!DİKKAT BONUS PAY KONTROL BANKADAN GELMİYOR....!!!!!!");
        Response.Write("<br></br>");
    }
    else
    {
        Response.Write("<br></br>");
        Response.Write("!!!!!!Onaylı işlem olmadığından kontrole gerek yok onay yoksa gphashdata gelmez!!!!!!");
        Response.Write("<br></br>");
    }
}
}
}

```

NOT: bu doğrulama akışı dışında mevcut kod yapınızda responseHashparams ile doğrulama yapıyorsanız o kısım içinde gerekli alanlar döndüğünden herhangi bir hataya neden olmayacaktır.

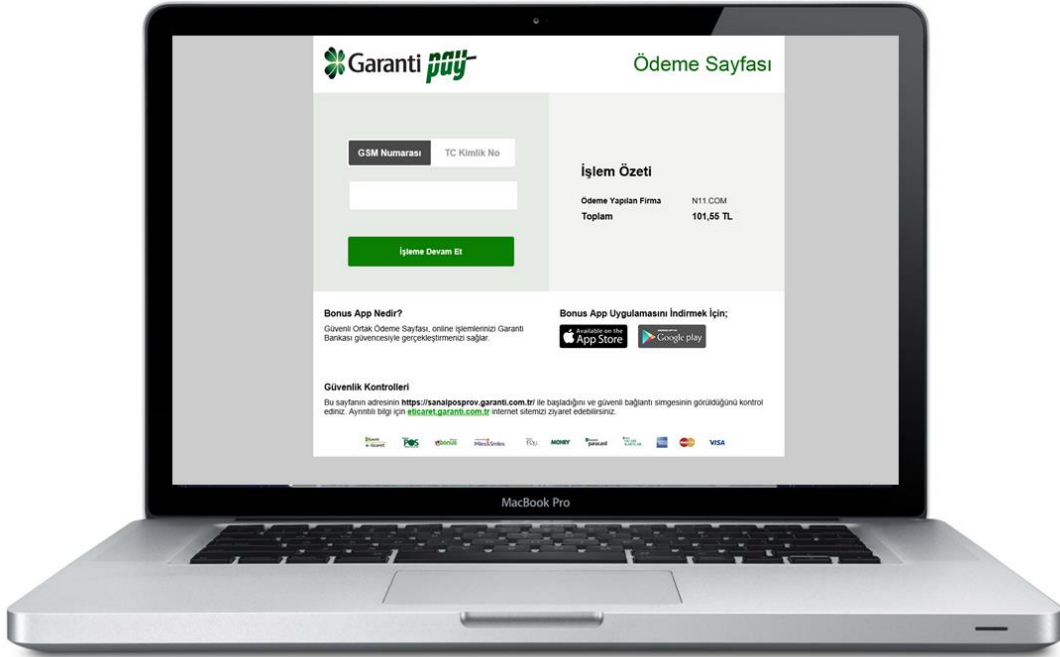
Örnek istek sayfası için aşağıda verilen sayfa dikkate alınabilir.



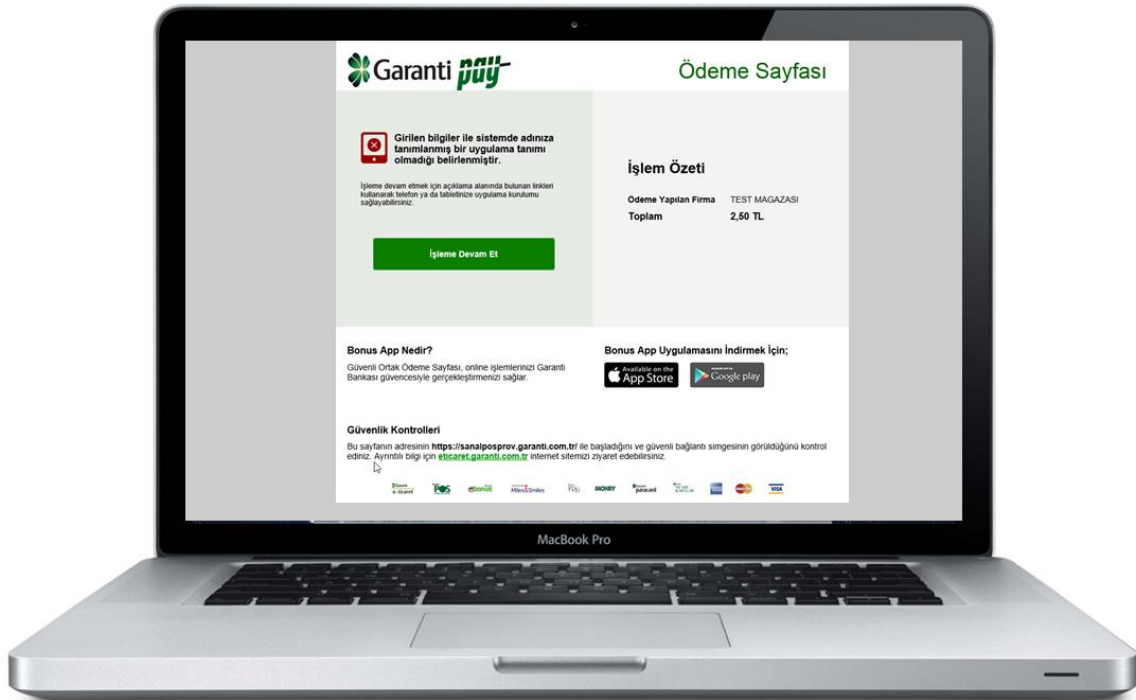
preprod_7000679_30690168_GarantiPaY_Test.html

BonusPay Ödeme Sayfası Görselleri

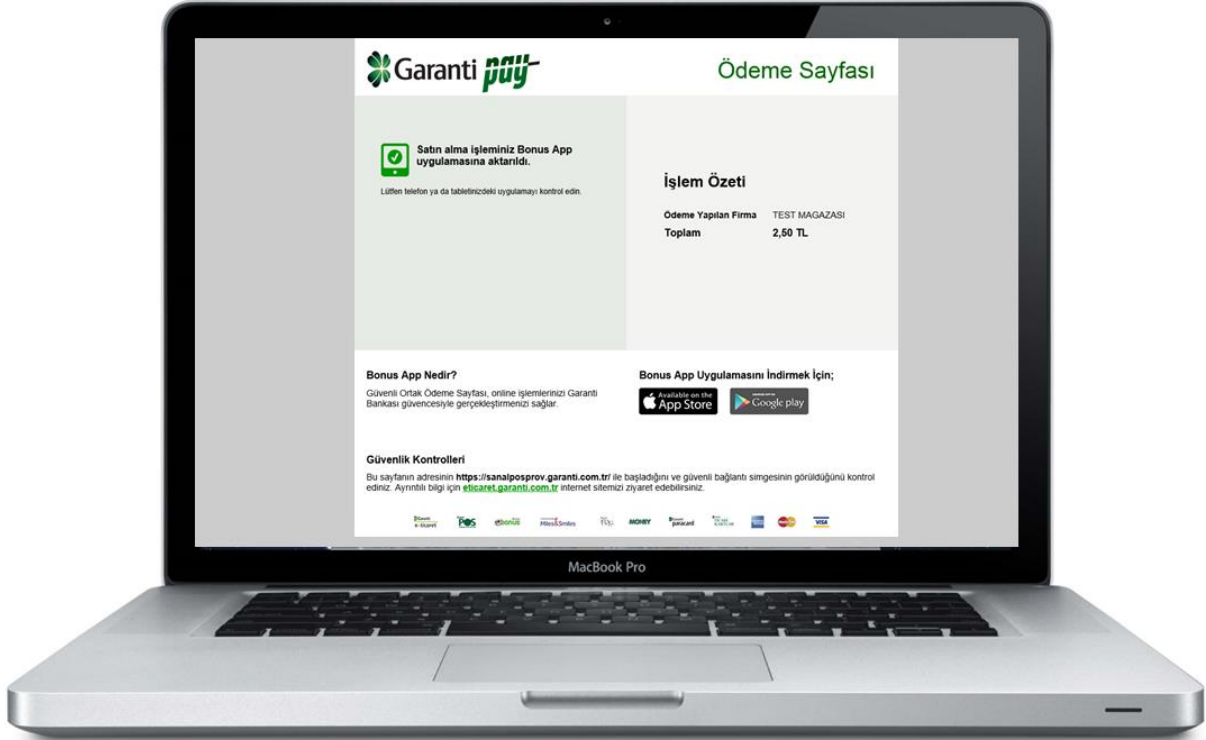
Kullanıcının TCKN ve/veya GSM numarasının istendiği ve ilk olarak görüntüleyeceği sayfa aşağıdaki gibi olacaktır.



Girilen TCKN ve/veya GSM numarası için sistemde bulunan müşteri numarasının bir uygulama tanımı yok ise kullanıcı ilk olarak Uygulama indirme sayfasına yönlendirilecektir.



Eğer kullanıcının bir uygulaması bulunuyorsa , sistem tarafından kullanıcıya bildirim gönderimi yapılırken web ekranında aşağıdaki yönlendirme sayfası gösterilecektir.



İşyeri Tarafından GarantiPay Mail Order İsteğinin Oluşturulması

Mail order işlem; kart sahibinin bulunmadığı bir ortamda kart sahibi adına başlatılan işlem türüdür. GarantiPay üzerinden başlatılan mail order işlemlerde işyeri tarafından kart sahibinin TCKN veya GSM numarası alınarak işlem başlatılır. Kullanıcı kendisine gelen notifi ile BonusFlaş uygulamasına girerek ya da notifi beklemeden direk BonusFlaş uygulamasına girerek kendisine yönlendirilen işlemi görüp işlemi provizyona dönüştürebilir.

İstek içerisinde gönderilen bir ürün bilgisi varsa ürün bilgisi müşteriye gösterilerek hangi ürüne yönelik provizyon gerçekleştirdiği detaylandırılabilir.

GarantiPay mail order istekler “İşyeri sanalpos XML entegrasyonu” yada “Sanalpos kullanıcı arayüzü- manuel işlemler ekranı” üzerinden yapılabilir.

XML ile gönderilen istek yapısı aşağıdaki gibi olacaktır.

```
<?xml version="1.0" encoding="UTF-8"?>
<GVPSRequest>
  <Mode>PROD</Mode>
  <Version>v0.01</Version>
  <ChannelCode></ChannelCode>
  <Terminal>
    <ProvUserID>PROVAUT</ProvUserID>
    <HashData>57C7FED5D9905C0F7F1A754270996CF3F5FE15DD</HashData>
    <UserID>XXXX</UserID>
    <ID>30690133</ID>
    <MerchantID>3424113</MerchantID>
  </Terminal>
  <Customer>
```



```

    <IPAddress>127.0.0.1</IPAddress>
    <EmailAddress>eticaret@garanti.com.tr</EmailAddress>
</Customer>
<Card>
    <Number></Number>
    <ExpireDate></ExpireDate>
    <CVV2></CVV2>
</Card>
<Order>
    <OrderID></OrderID>
    <GroupID></GroupID>
</Order>
<Transaction>
    <Type>gpdatarequest</Type>
    <SubType>sales</SubType>
    <InstallmentCnt></InstallmentCnt>
    <Amount>100</Amount>
    <CurrencyCode>949</CurrencyCode>
    <CardholderPresentCode>0</CardholderPresentCode>
    <ReturnServerUrl></ReturnServerUrl>
    <MotoInd>Y</MotoInd>
    <GarantiPaY>
        <bnsuseflag>N</bnsuseflag>
        <fbbuseflag>N</fbbuseflag>
        <chequeuseflag>N</chequeuseflag>
        <mileuseflag>N</mileuseflag>
        <CompanyName>Ozgur and Associates</CompanyName>
        <OrderInfo></OrderInfo>
        <TxnTimeOutPeriod>300</TxnTimeOutPeriod>
        <NotifSendInd>Y</NotifSendInd>
        <ReturnUrl></ReturnUrl>
        <TCKN>18007904436</TCKN>
        <GSMNumber>5354194523</GSMNumber>
        <InstallmentOnlyForCommercialCard>Y</InstallmentOnlyForCommercialCard>
        <TotalInstallmentCount>2</TotalInstallmentCount>
        <GPIInstallments>
            <Installment>
                <Installmentnumber>2</Installmentnumber>
                <Installmentamount>2234</Installmentamount>
            </Installment>
            <Installment>
                <Installmentnumber>4</Installmentnumber>
                <Installmentamount>1522</Installmentamount>
            </Installment>
        </GPIInstallments>
    </GarantiPaY>
</Transaction>
</GVPSRequest>

```

Bu istek içerisinde kırmızı ile belirtilen alanlar mail order isteği için önemli olan alanlardır.

<Type> alanında istek tipi “gpdatarequest” olarak yollanmalıdır. Tüm GarantiPay işlemlerinde kullanılan istek tipi bu şekildedir.

<MotoInd> alanında “Y” değerinin gönderilmesi sağlanmalıdır. Bu alan işlemin mail order işlem olduğunu gösteren alandır.

<ReturnServerUrl> alanı, mail order işlem sonucunun işyerinin vereceği bir server adresine yönlendirilmesi için kullanılır. Server üzerinden işyeri serverına yollanan işlem sonuç bilgileri işyeri tarafında işlem sonucunun bilinmesi ve provizyon sonrasında işyeri tarafındaki süreçlerin yürütülmesi amacı ile kullanılır. İşyeri server URL inin port bilgisinin 8543 olan bir sayfa adresi olması zorunludur. İşyerinin bu adresi için banka tarafında bir firewall tanımı yapılması zorunluluğu vardır. Bu özelliği kullanacak firmaların server IP lerine dışarıdan erişim yapılan IP bilgisini banka tarafı ile paylaşımları ve gerekli tanımı yaptırmaları gerekir.

<OrderInfo> alanında mail order işlemin hangi ürün için yapıldığı bilgisi yer alır. Bu bilginin BonusFlaş üzerinde ilgili işlem içerisinde kullanıcıya gösterilmesi sağlanır. Bu şekilde kullanıcının satın aldığı ürünle ilgili bilgilendirilmesi sağlanır.

<TxnTimeOutPeriod> alanında işlemin kullanıcı tarafında ne kadar süre aktif olarak görüntülenebileceği ve ne kadarlık bir sürede kullanıcının bu işlemi onaylayabileceği bilgisi bulunur. Bu alan sn cinsinden data gönderilmesi gereken bir alandır. Maksimum işlem süresi 1 gündür.

<NotifSendInd> alanında mail order işlemlerde notif gönderimi zorunludur. Bu alanda “Y” değerinin gönderilmesi gerekir.

<TCKN> alanında işlem yapan müşterinin TCKN bilgisi gönderilir. TCKN veya GSM in en az biri zorunludur. TCKN ile GSM aynı anda gönderilirse ilk olarak TCKN üzerinden kontroller yapılır. TCKN ile bir kullanıcı datası bulunursa bu kullanılır. TCKN geçerli bir TCKN değil ise GSM ile kullanıcı kontrollerine geçilir.

<GSMNumber> alanında TCKN gönderilmiyorsa kullanıcının GSM numarası yollanır.

Sanalpos kullanıcı ara yüzünde manuel işlemler ekranı üzerinden, GarantiPay Mail Order işlem gerçekleştirilebilir. İşlem tipi alanında “GarantiPay” seçiminin yapılması gereklidir.

İş Yeri : 7000670 35204-ÖDEMiŞ VERGİ DAİRESİ | Terminal : 30090168 35204-ÖDEMiŞ VERGİ DAİRESİ | Kullanıcı : CEM | Rol : İş Yeri Kullanıcısı | Çıkış

Giriş
Siparişler
İşlemler
Manuel İşlemler
Manuel İşlem Giriş
Toplu Sipariş Onayları
Ürünler
Günsonu
Yönetim
BonusPay
Arşiv Arama
Batch Rapor

Manuel İşlemler

Manuel İşlemler Giriş















Tüm alanları boşaltmak için "Temizle" düğmesine basın. Kriterlerinizi belirledikten sonra arama yapmak için "Ara" düğmesine basın.

İşlem Tipi : GarantiPay
Alt İşlem Tipi :

İşlem Bilgileri

Grup Numarası:		Taksit Seçeneği:	Peşin
Sipariş Numarası:		Döviz Kuru:	TL
Kredi Kartı Numarası:		Kullanılan Bonus:	
Son Kullanma Tarihi:	/	Kazanılan Bonus:	
CVV2:		Açıklama:	
Tutar:		İşlem Numarası:	
SMS Doğrulama Bilgisi :		Ekstre Doğrulama Bilgisi:	
Fraud Sebebiyle İptal:		İade Edilecek Çek Tutarı:	
İade Edilecek TL Tutarı:		İade Edilecek Player Tutarı:	
TCKN:		GSM Numarası:	
Kampanya Taksitini App Taksit Gösterimine Ekle:	Hayır	Ürün / Hizmet Bilgisi:	
App'te Bonus Kullanımına İzin Ver:	Evet	App'te Firma Bonus Kullanımına İzin Ver:	Evet
App'te Mil Kullanımına İzin Ver:	Evet	App'te Çek Kullanımına İzin Ver:	Evet
App'te Gösterimi Yapılacak Taksit Adedi:	0		

Temizle
Tamam

Tutar alanına peşin işlem tutarının girilmesi gereklidir. Peşin işlem tutarı zorunlu alandır.

TCKN ve GSM alanına kullanıcının TCKN veya GSM bilgilerinden en az birisinin girilmesi gereklidir.

“Kampanya Taksitini App Taksit Gösterimine Ekle” alanında eğer işyeri tarafında bir bildirimsiz kampanya tanımı varsa ve müşteriye kampanyadan kazanılan taksit işyerinin bir opsiyonu gibi sunuluyorsa, bu alanda Evet seçimi yapılarak gönderim yapılmalıdır.

“Ürün / Hizmet Bilgisi” alanında satın alınan ürün bilgisi yollanmalıdır.

“App'te Bonus Kullanımına İzin Ver” alanında eğer işyeri işlemde Bonus kullanılmasına izin veriyorsa “Evet” seçimi ile işlem sonlandırılmalıdır.

“App'te Firma Bonus Kullanımına İzin Ver” alanında işyeri tarafında firma bonus tanımı varsa ve bu özellik kullanıcıya App üzerinden bir seçenek olarak sunulmak isteniyorsa “Evet” seçimi ile işlem yollanmalıdır.

“App'te Mil Kullanımına İzin Ver” alanında işyeri tarafından MIL kullandırımı yapılmak isteniyorsa bu değerin yollanması sağlanmalıdır. Mevcutta bu özellik şu an için APP üzerinde desteklenmemektedir.

“App'te Çek Kullanımına İzin Ver” alanında işyeri tarafından ÇEK kullandırımı yapılmak isteniyorsa bu değerin yollanması sağlanmalıdır. Mevcutta bu özellik şu an için APP üzerinde desteklenmemektedir.

“App'te Gösterimi Yapılacak Taksit Adedi” alanında işlemde kaç farklı taksit seçeneği sunulacağı bilgisi yer alacaktır. Bu alana girilen sayı kadar taksit bilgilerinin girileceği aşağıdaki alanların gösterimi yapılır.

“App'te Gösterilecek Taksit Numarası” alanında kullanıcıya ekranda gösterilecek olan taksitin sayısı yer alır. Örnek kullanıcıya 2 4 6 şeklinde taksitler gösterilecekse ilk taksit bilgisi için bu alanda “2” ifadesi yer alır.

“App'te Gösterilecek Taksit Tutarı” girilen taksit sayısı için alınması istenen tutar bu alanda yollanılır.

“Ödül Kullanımlarında Taksit Tutarını Yeniden Hesaplama Oranı” eğer bonus kullanımında , Bonus sonrasında karttan çekilecek tutar için tekrar bir hesaplama yapılması isteniyor ise yeniden hesaplamada kullanılacak taksit oranı bu alanda gönderilir .

GarantiPay Bekleyen İşlemin İptalinin Yapılabilmesi

GarantiPay işlemlerinde Garanti Pay ödeme sayfasından ya da App üzerinde işlem listeleme ekranından iptal isteği gelebilecektir. Aynı şekilde timeout a düşen işlemlerde GarantiPay sayfası üzerinden bir iptal isteği başlatılarak işyerine bu işlemler için hatalı işlem cevabının dönülmesi sağlanacaktır.

İptal isteklerinde işlem tipi alanında “garantipaycancel” ifadesi gönderilmelidir.

Transaction alanında garantipay tagi içerisinde konulacak “status” alanında iptal sebebi iletilecektir. Timeout sebebiyle yapılan işlemde “E” , müşteri isteği ile yapılacak iptal isteğinde “C” bilgisinin gönderilmesi gereklidir.

Ek olarak GarantipayID nin garanti pay alanı altında GPID alanında istek mesajına eklenmesi zorunludur.

Order id alanında iptali yapılan işlemin sipariş numarası bulunmalıdır. Sipariş numarası ile GarantipayID nin kontrolü yapılarak, eğer sipariş numarası ile iptal edilen GarantiPay ID farklı ise “Sipariş numarası ile GarantiPayID uyumsuzdur.” Hatası verilir.

İptal isteğinin mesaj yapısı

```
<?xml version="1.0" encoding="UTF-8"?>
<GVPSRequest>
  <Mode>PROD</Mode>
  <Version>v0.01</Version>
  <ChannelCode></ChannelCode>
  <Terminal>
    <ProvUserID>PROVAUT</ProvUserID>
    <HashData>57C7FED5D9905C0F7F1A754270996CF3F5FE15DD</HashData>
    <UserID>XXXX</UserID>
    <ID>30690133</ID>
    <MerchantID>3424113</MerchantID>
  </Terminal>
  <Customer>
    <IPAddress></IPAddress>
    <EmailAddress></EmailAddress>
  </Customer>
  <Card>
    <Number></Number>
    <ExpireDate></ExpireDate>
    <CVV2></CVV2>
  </Card>
  <Order>
    <OrderID>122132131212423</OrderID>
    <GroupID></GroupID>
  </Order>
  <Transaction>
    <Type>garantipaycancel</Type>
```

```

<SubType></SubType>
<InstallmentCnt></InstallmentCnt>
<Amount>100</Amount>
<CurrencyCode>949</CurrencyCode>
<CardholderPresentCode>0</CardholderPresentCode>
<MotoInd>N</MotoInd>
<GarantiPaY>
  <GPID>12124325487518475</ GPID >
  <Status>E</Status >
</GarantiPaY>
</Transaction>
</GVPSRequest>

```


Şeklinde olacaktır.

GPID ve Statü bilgisi yeni oluşturulacak işlem tipi ile SanalPoS tarafından Host tarafında provizyona aktarılacaktır.


GarantiPay Ödeme Sayfasında İptal Butonunun ve Time Out Süresi

GarantiPay ödeme sayfasının bekleme sayfasına “kalan süre” alanının ve “İşlemi İptal Et” botunun konulması sağlanmıştır.

Ekran görüntüsü aşağıda yer almaktadır.



Ödeme Sayfası



Satın alma işleminiz BonusFlaş uygulamasına aktarıldı.

Lütfen telefon ya da tabletinizdeki uygulamayı kontrol ediniz.

Kalan Süre

5:30

İşlemi İptal Et

İşlem Özeti

Ödeme Yapılan Firma	TEST MAGAZASI
Toplam	2,50 TL

Kalan süre alanında eğer istek içerisinde TxnTimeOutTime alanında gelen bir süre varsa bu süreden geri sayım yapılmaktadır. Eğer bir süre yoksa geri sayım yapılan süre default süre olarak değerlendirilen 300 sn üzerinden olacaktır.

Kalan süre alanı 0 olduğunda işlem için beklemede olan işlemin iptali yönelik aksiyon alınır. İşyeri tarafına iptal olan işlem için iptal sebebine uygun olarak cevap dönülür.

İptal cevabı sanalPoS hata mesajı olarak 92 hatası ile dönülecektir.

Timeout durumunda cevap mesajı “İşlem süresi doldu” , müşteri isteği ile iptal durumunda” Müşteri isteği ile iptal” geri dönüşü yapılacaktır.

Sipariş Sorgu Cevabına Bekleyen İşlemin ve İptal İşleminin Gösterilmesi

Mevcut durumda sipariş sorgu isteğinde eğer sadece garantipay kayıt atma isteği varsa sonuç boş dönmektedir.

Bu proje ile eğer işlem kayıt atma statüsünde ise bu durumun order tablosundan kontrol edilerek , provizyon bekleniyor (waiting) açıklaması ile geri döndürülmesi sağlanmalıdır.

```
<OrderInqResult>
<ChargeType></ChargeType>
<PreAuthAmount></PreAuthAmount>
<PreAuthDate/>
<AuthAmount>1000</AuthAmount>
<AuthDate></AuthDate>
<RecurringInfo></RecurringInfo>
<RecurringStatus/>
<Status>WAITING</Status>
<RemainingBNSAmount>0</RemainingBNSAmount>
<UsedFBBAmount>0</UsedFBBAmount>
<UsedChequeType/>
<UsedChequeCount>0</UsedChequeCount>
<UsedChequeAmount>0</UsedChequeAmount>
<UsedBnsAmount>0</UsedBnsAmount>
<InstallmentCnt>0</InstallmentCnt>
<CardNumberMasked></CardNumberMasked>
<CardRef/>
<Code></Code>
<ReasonCode></ReasonCode>
<SysErrMsg/>
</OrderInqResult>
```

GarantiPay OOS üzerinden yapılan sorgulamalarda waiting statüsünde olan işlemlerin sorgulamasına devam edilmelidir. Bu durumda işlem için işyerine bir dönüş yapılmamalıdır.

İptal olan isteklerde sorgu cevabı işlem statüsü CANCEL olarak dönecektir.

```
<OrderInqResult>
<ChargeType></ChargeType>
<PreAuthAmount></PreAuthAmount>
<PreAuthDate/>
<AuthAmount>1000</AuthAmount>
<AuthDate></AuthDate>
<RecurringInfo></RecurringInfo>
<RecurringStatus/>
<Status>CANCEL</Status>
<RemainingBNSAmount>0</RemainingBNSAmount>
<UsedFBBAmount>0</UsedFBBAmount>
<UsedChequeType/>
```

```
<UsedChequeCount>0</UsedChequeCount>
<UsedChequeAmount>0</UsedChequeAmount>
<UsedBnsAmount>0</UsedBnsAmount>
<InstallmentCnt>0</InstallmentCnt>
<CardNumberMasked>979203*****2898</CardNumberMasked>
<CardRef/>
<Code></Code>
<ReasonCode></ReasonCode>
<SysErrMsg/>
</OrderInqResult>
```

İşyeri App üzerinden Satın Alma Öncesinde BonusFlaş Varlık Kontrolü

Aşağıdaki linkler ile merchant uygulamasında check-out'a gelmeden önce kullanıcının cihazında BonusFlaş'ın yüklü olup olmadığını öğrenebilme imkanı bulunmaktadır. Aşağıdaki URL'ler uygulama varlık kontrolü ve desteklenen platform bilgileri (BonusFlaş'ı download'a yönlendirebilmek için gerekli min OS Version bilgileri) için kullanılabilir. IOS tarafında uygulama varlık kontrolü adına verilen link Deep Link, ancak BonusFlaş uygulamasını çağırmak için SanalPos üzerinden dönecek link universal link'tir.

"Linklerin dinamik olarak kullanılması, ilerde linke yapılacak bir ekleme olma ihtimaline karşı (GarantiPay'in Cep Şube , İgaranti platformlarında da desteklenme durumu gibi) önerilir. IOS 9 ile beraber gelen durumla ilgili (App Store'a uygulama iletilirken, uygulama içinden açılacak linkleri vermek gerektiği için) "iosDeepLink": "iosGarantiBonusFlas://" kısmındaki ios linkini statik olarak kod içine gömerek devam etmek gerekse de ana linkin dinamik kullanımı önerilir.

PreProd Linki:

<https://sanalposprotest.garanti.com.tr/interface/GarantiPayAppList.json>

```
[
  {
    "id": "bonusflas",
    "app": {
      "androidPackageName": "com.garanti.bonusapp.tst",
      "androidMinVersion": "16",
      "iosDeepLink": "iosGarantiBonusFlas://",
      "iosMinVersion": "9.0"
    }
  }
]
```

Mevcut Prod Linki:

<https://sanalposprov.garanti.com.tr/interface/GarantiPayAppList.json>

```
[
  {
    "id": "bonusflas",
    "app": {
      "androidPackageName": "com.garanti.bonusapp",
      "androidMinVersion": "16",
      "iosDeepLink": "iosGarantiBonusFlas://",
      "iosMinVersion": "9.0"
    }
  }
]
```

```

    }
  }
]

```

İşyeri App üzerinden Satın Alma İsteğinin Oluşturulması

İşyeri ilk olarak SanalPoS XML servisini çağırarak işlem datasının sanalPoS tarafına kayıt edilmesini sağlayacaktır. Bunun için `<Type>gpdatarequest</Type>` istek tipi ile bir istek yollanılacaktır.

Provizyon işleminde kullanılması istenilen istek tipi SubType alanında yollanılacaktır. `<SubType>sales</SubType>` şeklinde kullanım gerçekleşecektir. Bu alanda sales ve preauth işlem tipleri gönderilebilecektir.

İşlem oluşturulurken aşağıdaki alanlar belirtildiği şekilde kullanılmalıdır.

MotoInd = N
NotifSendInd = N

Bu istek içerisinde WEB akışlarında anlatılan dataların XML yapısında sağlanan alanlarda yollanması sağlanacaktır. Gpdatarequest işlem tipi için aşağıda dataların ilgili alanda yollanması sağlanmalıdır.

```

<GarantiPaY>
  <bnsuseflag></bnsuseflag>
  <fbbuseflag></fbbuseflag>
  <chequeuseflag></chequeuseflag>
  <miluseflag></miluseflag>
  <CompanyName></CompanyName>
  <ReturnUrl></ReturnUrl> // işyeri App ine cevap mesajının yollanılacağı adres burada yer alacaktır.
  <TCKN></TCKN> // Varsa yollanılacak
  <GSMnumber></GSMnumber> // Varsa yollanılacak
  <TotalInstamenlCount>N</TotalInstamenlCount>
  <InstallmentOnlyForCommercialCard>Y</InstallmentOnlyForCommercialCard>
  <TxnTimeOutPeriod>300</TxnTimeOutPeriod>
  <GPInstallments>
    </Installment>
    <Instalmentnumber>N</Instalmentnumber>
    <Instalmentamount>TTTTKK</Instalmentamount> // T-Tutar K-Kuruş Örn:120.20 TL için 12020
    <Installmentratewithreward>YYYOO</Installmentratewithreward> // Y-Yüzden O-Ondalık Değer
    Örn: %4,20 için 420
    <Installment>
    ...
    </Installment>
    <Instalmentnumber>N</Instalmentnumber>
    <Instalmentamount>TTTTKK</Instalmentamount>
    <Installmentratewithreward>YYYOO</Installmentratewithreward>
    <Installment>
  </GPInstallments>
</GarantiPaY>

```

Oluşturulan istek

Test ortamı için

<https://sanalposprotest.garanti.com.tr/VPServlet> adresine yollanılmalıdır.

Test ortamında kullanılacak mağaza bilgileri

<ID>30690133</ID>

<MerchantID>3424113</MerchantID>

Şeklinde. PROVAUT kullanıcısına ait şifre bilgisi 123qweASD/ şeklindedir.

Prod ortamı için

<https://sanalposprov.garanti.com.tr/VPServlet>

adresini kullanacaktır. Burada işyerinin kendi işyeri ve terminal bilgileri kullanılacaktır.

İstek cevabında <GPID>XXXXXXXXXX</GPID> değeri sanalPos cevabında dönülecektir. Bu data ile aşağıda anlatılan akış ile banka APP çağırımı sağlanacaktır.

Çağrılacak APP linkleri aşağıdaki gibi sanalPos İstek Cevabında döndürülecektir.

```
<AppLinks>
  <Link>
    <ID>1</ID>
    <AndroidPackageName>com.garanti.bonusapp.tst</AndroidPackageName>
    <AndroidClassName>com.garanti.bonusapp.activity.start.StartActivity</AndroidClassName>
    <IOSDeepLink>https://bonusgw-tst.garanti.com.tr/garantiPay</IOSDeepLink>
  </Link>
  <Link>
    <ID>2</ID>
    <AndroidPackageName>com.garanti.cepsubesi</AndroidPackageName>
    <AndroidClassName/>
    <IOSDeepLink>//cepsube</IOSDeepLink>
  </Link>
  <Link>
    <ID>3</ID>
    <AndroidPackageName>com.igaranti</AndroidPackageName>
    <AndroidClassName/>
    <IOSDeepLink>//igaranti</IOSDeepLink>
  </Link>
</AppLinks>
```

IOS için Banka App Çağırımı

Gelen cevapta bulunan app linklerini sırayla açmayı denenecektir.

Test : <https://bonusgw-tst.garanti.com.tr/garantiPay?garantiPayId=xxx&merchantId=yyy&terminalId=zzz>

Prod: <https://bonusappgw.garanti.com.tr/garantiPay?garantiPayId=xxx&merchantId=yyy&terminalId=zzz>

Örnek app linki:

[https://bonusgw-](https://bonusgw-tst.garanti.com.tr/garantipay?garantiPayId=30690133511011683605A&merchantId=3424113&terminalId=30690133)

[tst.garanti.com.tr/garantipay?garantiPayId=30690133511011683605A&merchantId=3424113&terminalId=30690133](https://bonusgw-tst.garanti.com.tr/garantipay?garantiPayId=30690133511011683605A&merchantId=3424113&terminalId=30690133)

Bonus flaş uygulamasında işlem tamamlandığında Merchant app

orderId=xxxxxx

değeri ile geri çağırılır

Aşağıda örnek Objective C kodu paylaşıyorum.

Sanalpos Request sonrası bonus app uygulamasının açılması

Sanal POS response datasından IOSDeepLink alanından çağırılacak URL bilgisi alınır

```
<GPID>00505181602619627493723143</GPID>
<AppLinks>
  <Link>
    <ID>1</ID>
    <AndroidPackageName>com.garanti.bonusapp.tst</AndroidPackageName>
    <AndroidClassName>com.garanti.bonusapp.activity.start.StartActivity</AndroidClassName>
    <IOSDeepLink>https://bonusgw-tst.garanti.com.tr/garantipay</IOSDeepLink>
  </Link>
</AppLinks>
</Transaction>
</GVPSResponse>
```

```
if ([responseCode isEqualToString:@"00"]) {
```

```
    // URL oluşturulur
```

```
    NSString *customURL = [NSString
stringWithFormat:@"%s?garantiPayId=%s&merchantId=%s&terminalId=%s", appLink, gpID, merchantID, terminalID];
    //@"https://bonusgw-
tst.garanti.com.tr/garantipay?garantiPayId=30690133511011683605A&merchantId=3424113&terminalId=30690133";
```

```
    //Oluşan URL in gösterdiği uygulamanın (BonusApp) yüklü olup olmadığı kontrol edilir.
```

```
    if ([[UIApplication sharedApplication]
canOpenURL:[NSURL URLWithString:customURL]])
```

```
    {
        //Bonus App yüklüyse bu uygulamaya geçiş yapılır.
        [[UIApplication sharedApplication] openURL:[NSURL URLWithString:customURL]];
    }
```

```
    else
    {
```

```
        //Bonus app yüklü değilse uygun mesaj gösterilir
```

```
        UIAlertView *alert = [[UIAlertView alloc] initWithTitle:@"URL error"
message:[NSString stringWithFormat:
@"No custom URL defined for %s", customURL]
delegate:self cancelButtonTitle:@"Ok"
otherButtonTitles:nil];
```

```
        [alert show];
    }
```

```
} else {
```

```
    //İşlem başarılı değilse uygun mesaj gösterilir
```

```
    UIAlertView *alert = [[UIAlertView alloc] initWithTitle:@"Error in provision"
message:response
delegate:self cancelButtonTitle:@"Ok"
otherButtonTitles:nil];
```

```
    [alert show];
```

```
}
```

Bonus App ödeme işleminin sonucunun Merchant uygulama tarafından alınması

```
- (BOOL)application:(UIApplication *)application openURL:(NSURL *)url
sourceApplication:(NSString *)sourceApplication annotation:(id)annotation
{
    NSLog(@"Calling Application Bundle ID: %@", sourceApplication);
    NSLog(@"URL scheme:%@", [url scheme]);
    NSLog(@"URL query: %@", [url query]);

    NSString *sQuery = [url query];
    if([sQuery length] > 0)
    {
        NSMutableDictionary *params = [[NSMutableDictionary alloc] init];
        for (NSString *param in [sQuery componentsSeparatedByString:@"&"]) {
            NSArray *elts = [param componentsSeparatedByString:@"="];
            if([elts count] < 2) continue;
            [params setObject:[elts objectAtIndex:1] forKey:[elts objectAtIndex:0]];
        }
        //OrderId değerinin alınması
        NSString* orderId= [params objectForKey:@"orderId"];
    }

    return YES;
}
```

Android için Banka App Çağırımı

Not: **Android** GarantiPay akışlarında BonusFlaş açılırken, **top-level launcher** üzerinden tetiklenme gerekmektedir.

Sanalpos Request sonrası bonus app uygulamasının açılması

Sanal POS response datasından AndroidPackageName ve AndroidClassName alanından çağırılacak app bilgisi alınır

```
<GPID>00505181602619627493723143</GPID>
<AppLinks>
  <Link>
    <ID>1</ID>
    <AndroidPackageName>com.garanti.bonusapp.tst</AndroidPackageName>
    <AndroidClassName>com.garanti.bonusapp.activity.start.StartActivity</AndroidClassName>
    <IOSDeepLink>https://bonusgw-tst.garanti.com.tr/garantipay</IOSDeepLink>
  </Link>
</AppLinks>
</Transaction>
</GVPSResponse>
```

Bonus flaş uygulamasında işlem tamamlandığında Merchant app
orderId=xxxxxx
değeri ile geri çağırılır.

İşyeri tarafından cevap mesajında dönen OrderID bilgisi ile orderinq istişinin başlatılarak işlem sonucuna ait bilgilerin sanalpos üzerinden alınması sağlanmalıdır.

OrderInq istek mesajının cevabında sorgulanan sipariş altındaki en son provizyona denemesine (satış veya ön otorizasyon) ait işlem için, işlem detayı kontrol edilerek

- CardRef
- Code
- ReasonCode
- SysErrMsg

Bilgilerinin de cevap mesajında döndürülmesi sağlanacaktır.

OrderInqResult alanında yeni alanlar aşağıdaki gibi gözükecektir.

```
<Order>
  <OrderID>SIST412178D8D8724AE0A1FC9C008EE9F3C0</OrderID>
  <GroupID/>
<OrderInqResult>
  <ChargeType>S</ChargeType>
  <PreAuthAmount>0</PreAuthAmount>
  <PreAuthDate/>
  <AuthAmount>1110</AuthAmount>
  <AuthDate>2016-03-18 20:34:36.929</AuthDate>
  <RecurringInfo>N</RecurringInfo>
  <RecurringStatus/>
  <Status>APPROVED</Status>
  <RemainingBNSAmount>0</RemainingBNSAmount>
  <UsedFBBAmount>0</UsedFBBAmount>
  <UsedChequeType/>
  <UsedChequeCount>0</UsedChequeCount>
  <UsedChequeAmount>0</UsedChequeAmount>
  <UsedBnsAmount>0</UsedBnsAmount>
  <InstallmentCnt>0</InstallmentCnt>
  <CardNumberMasked>554960*****1023</CardNumberMasked>
  <CardRef/>
  <Code>00</Code>
  <ReasonCode>00</ReasonCode>
  <SysErrMsg/>
</OrderInqResult>
</Order>
```

App tarafında dönecek cevap için Örnek JAVA kodu

```
/* Note that garantiPayAppList is a sample wrapper list keeping application details
belonging to Garanti Bank.
```

```
* This must be populated with respect to Garanti VirtualPos response XML.
```

```
*/
```

```
if(responseCode.equals("00")) {
    try {

        // Create GarantiPay intent

        Intent intentGarantiPay = new Intent( Intent.ACTION_MAIN );
        intentGarantiPay.setComponent( new ComponentName( androidPackageName,
androidClassName ) );

        // Populate parameters into a bundle
        Bundle bundle = new Bundle( );
```

```

        bundle.putString( "garantiPayId", gpId );
        bundle.putString( "merchantId", merchantID );
        bundle.putString( "terminalId", terminalID );

        intentGarantiPay.putExtras(bundle);

// Force it to open in a top-level launcher (as a separate application in the launcher)

        intentGarantiPay.addCategory(Intent.CATEGORY_LAUNCHER);
        intentGarantiPay.setFlags(Intent.FLAG_ACTIVITY_NEW_TASK |
Intent.FLAG_ACTIVITY_CLEAR_TOP);
        startActivity( intentGarantiPay );

    } catch(Exception e) {
        new AlertDialog.Builder(GarantiPayActivity.this)
            .setTitle("Hata")
            .setMessage("Bonus Flas Bulunamadi")
            .setPositiveButton(android.R.string.yes, new
DialogInterface.OnClickListener() {
                public void onClick(DialogInterface dialog, int which) {

                }

            })

            .setIcon(android.R.drawable.ic_dialog_alert)
            .show();
    }
} else {
    new AlertDialog.Builder(GarantiPayActivity.this)
        .setTitle("Hata")
        .setMessage(xml)
        .setPositiveButton(android.R.string.yes, new
DialogInterface.OnClickListener() {
            public void onClick(DialogInterface dialog, int which) {

            }

        })

        .setIcon(android.R.drawable.ic_dialog_alert)
        .show();
    }
}
}

```

İşlem sonucunda order ID değerinin alınması

```

@Override

protected void onResume()

{

    super.onResume();

    Uri data = getIntent().getData();

    if (data != null) {

```

```
orderId.setText (data.getQueryParameter ("orderId" ) );  
}  
}
```

Test ortamında data atma için kullanılabilecek örnek dosya aşağıda ekte yer almaktadır.



preprod - DataRequest.html

Destek için eticaret@garanti.com.tr veya 444 0 339 / 7