**🛒 1. Fiziksel POS Cihazında Ödeme Süreci**

(Bir restoranda veya mağazada kredi kartıyla ödeme yapılırken olan süreç)

1. **POS cihazına tutar girilir.**
   * Manuel olarak veya restoran yazılımı ile entegre şekilde otomatik gönderilir.
2. **Kart, POS cihazına takılır, okutulur veya temassız kullanılır.**
   * Chip, manyetik şerit veya NFC ile işlem yapılabilir.
3. **POS, bankanın ödeme altyapısına bağlanır.**
   * Bankanın ödeme sistemine giderek işlemi doğrular.
4. **Banka işlemi onaylar veya reddeder.**
   * Yetersiz bakiye, şüpheli işlem vb. durumlar kontrol edilir.
5. **Ödeme tamamlanırsa fiş yazdırılır.**

**Entegrasyon için:**  
POS cihazlarının genellikle **REST API veya özel SDK’ları** vardır. Ödeme tutarını doğrudan POS’a göndermek için cihazın marka/modeline göre entegrasyon yapılır.

**💳 2. Sanal POS ile Online Ödeme Süreci**

(Örneğin, bir restoranın web sitesinden kredi kartıyla ödeme yapılırken olan süreç)

**📌 Adım Adım Süreç:**

1. **Kullanıcı ödeme sayfasına gelir, kart bilgilerini girer.**
   * Kart numarası, CVV, son kullanma tarihi gibi bilgiler girilir.
2. **Ödeme API’sine istek gönderilir.**
   * Restoranın sisteminden ödeme sağlayıcısına (PayTR, İyzico, Stripe, Garanti, İş Bankası vb.) **HTTPS POST isteği** gönderilir.
3. **Sanal POS, bankaya isteği iletir.**
   * Ödeme sağlayıcı, bankanın altyapısına kart bilgilerini gönderir ve provizyon alır.
4. **3D Secure doğrulaması yapılır (Bazı işlemler için zorunlu).**
   * Kullanıcının telefonuna **banka tarafından SMS ile tek kullanımlık şifre (OTP)** gelir.
5. **Banka işlemi onaylar veya reddeder.**
   * Kart limitine, sahtekarlık riskine vb. göre değerlendirme yapılır.
6. **Ödeme tamamlanır, kullanıcıya başarılı/başarısız sonucu gösterilir.**
   * Ödeme başarılıysa sipariş onaylanır.

**🛠️ Sanal POS Entegrasyonu Nasıl Yapılır?**

Eğer Django ile bir ödeme entegrasyonu yapacaksan, aşağıdaki yöntemlerden birini kullanabilirsin:

**1️⃣ PayTR (Türkiye’de popüler)**

* **Kolay API entegrasyonu** ve **3D Secure desteği** var.
* Rest API ile ödeme başlatıp sonucu kontrol edebilirsin.
* 🎯 **Döküman:** https://dev.paytr.com/

**2️⃣ İyzico**

* Türkiye’de yaygın kullanılan bir diğer servis.
* Django/Python ile entegre edilebiliyor.
* 🎯 **Döküman:** https://dev.iyzico.com/

**3️⃣ Stripe (Global için iyi bir seçenek)**

* Dünyada en çok kullanılan ödeme altyapılarından biri.
* 🎯 **Döküman:** <https://stripe.com/docs>

**🚀 Örnek Django Ödeme Entegrasyonu (Stripe ile)**

Eğer **Stripe kullanmak istersen**, basit bir ödeme entegrasyonu şöyle olabilir:

**1️⃣ Stripe Kütüphanesini Yükle**

bash

KopyalaDüzenle

pip install stripe

**2️⃣ Django View (Ödeme İşlemi)**

python

KopyalaDüzenle

import stripe

from django.conf import settings

from django.http import JsonResponse

stripe.api\_key = settings.STRIPE\_SECRET\_KEY # API Key'i settings.py'den al

def process\_payment(request):

try:

# Ödeme oluştur

payment\_intent = stripe.PaymentIntent.create(

amount=5000, # Ödeme tutarı (Kuruş cinsinden -> 50 TL = 5000)

currency="try",

payment\_method\_types=["card"], # Kredi Kartı ile ödeme

)

return JsonResponse({"clientSecret": payment\_intent["client\_secret"]})

except stripe.error.StripeError as e:

return JsonResponse({"error": str(e)}, status=400)

**3️⃣ Frontend’de Kullanım (JavaScript)**

javascript

KopyalaDüzenle

const stripe = Stripe("STRIPE\_PUBLIC\_KEY");

async function pay() {

const response = await fetch("/process-payment/");

const data = await response.json();

const { error } = await stripe.confirmCardPayment(data.clientSecret, {

payment\_method: {

card: cardElement,

billing\_details: { name: "Müşteri Adı" },

}

});

if (error) {

alert("Ödeme Başarısız: " + error.message);

} else {

alert("Ödeme Başarılı!");

}

}

**📌 Özet:**

✅ **Fiziksel POS’lar için cihazın API’si veya SDK’sı ile entegrasyon gerekir.**  
✅ **Sanal POS için ödeme sağlayıcısının (PayTR, İyzico, Stripe vb.) API’si kullanılır.**  
✅ **Django ile Stripe veya PayTR gibi bir sistemle kolayca entegre olabilirsin.**  
✅ **3D Secure ve güvenlik kontrolleri önemli.**

Eğer **hangi ödeme sistemini kullanacağını netleştirirsen**, daha spesifik bir entegrasyon adımı önerebilirim. 😎