• Introduction to integrating payment gateways (like Paytm) in Django projects.

Integrating payment gateways like Paytm into your Django project allows you to process online transactions securely and efficiently. Here’s a step-by-step guide to get you started:

**1. Prerequisites**

* **Django Installed**: Ensure you have Django installed in your project.
* **Paytm Merchant Account**: You'll need a Paytm Merchant Account to get the necessary API credentials (Merchant ID, Merchant Key, etc.).

**2. Setting Up Your Project**

Create a new Django project and app if you haven't already:

django-admin startproject myproject

cd myproject

python manage.py startapp payment\_app

**3. Install Required Libraries**

Install the necessary libraries for making HTTP requests and generating checksums:

pip install requests

pip install paytm-pg

pip install paytmchecksum

**4. Configure Paytm Credentials**

In your project’s settings.py file, configure your Paytm API credentials:

* python

# myproject/settings.py

PAYTM\_MID = "YOUR\_MID\_HERE"

PAYTM\_KEY = "YOUR\_MERCHANT\_KEY" # 16, 24, or 32 bytes AES key

PAYTM\_WEBSITE = "YOUR\_WEBSITE\_NAME"

PAYTM\_CLIENT\_ID = "YOUR\_CLIENT\_ID\_HERE"

PAYTM\_CALLBACK\_URL = "YOUR\_CALLBACK\_URL"

**5. Create Payment Views**

In your payment\_app/views.py file, create views for initiating payments, checking payment status, and handling payment responses:

* python

import logging

import requests

import json

from django.conf import settings

from django.http import HttpResponse

from django.shortcuts import render

from paytmpg import EnumCurrency, EChannelId, UserSubWalletType

from paytmpg import ExtendInfo, ShippingInfo, GoodsInfo, UserInfo, PaymentMode, Money

from paytmpg import PaymentDetailsBuilder, PaymentStatusDetailBuilder

from paytmpg import RefundDetailBuilder, RefundStatusDetailBuilder

from paytmpg import MerchantProperty, LibraryConstants

from paytmpg import Payment, Refund

from paytmchecksum import PaytmChecksum

logger = logging.getLogger(\_\_name\_\_)

def initiate\_payment(request):

# Your initialization code here

paytmParams = {

"MID": settings.PAYTM\_MID,

"ORDERID": "UNIQUE\_ORDER\_ID", # Replace with your unique order ID

"TXN\_AMOUNT": "1.00", # Replace with the transaction amount

"CUST\_ID": "CUSTOMER\_ID", # Replace with the customer’s ID

"MOBILE\_NO": "CUSTOMER\_MOBILE\_NO",

"EMAIL": "CUSTOMER\_EMAIL\_ID",

"INDUSTRY\_TYPE\_ID": "Retail",

"WEBSITE": settings.PAYTM\_WEBSITE,

"CHANNEL\_ID": "WEB",

"CALLBACK\_URL": settings.PAYTM\_CALLBACK\_URL,

}

# Generate the checksum using PaytmChecksum library

checksum = PaytmChecksum.generate\_checksum(paytmParams, settings.PAYTM\_KEY)

paytmParams['CHECKSUMHASH'] = checksum

response = requests.post(settings.PAYTM\_GATEWAY\_URL, data=paytmParams)

return HttpResponse(response.content)

**6. Create URLs**

Map your views to URLs in payment\_app/urls.py:

* python

from django.urls import path

from . import views

urlpatterns = [

path('initiate\_payment/', views.initiate\_payment, name='initiate\_payment'),

]

**7. Create Templates**

Create templates for your payment page and callback page:

**payment\_page.html**:

* html

<form method="post" action="{% url 'initiate\_payment' %}">

{% csrf\_token %}

<input type="hidden" name="ORDERID" value="UNIQUE\_ORDER\_ID">

<input type="hidden" name="TXN\_AMOUNT" value="1.00">

<input type="hidden" name="CUST\_ID" value="CUSTOMER\_ID">

<input type="hidden" name="MOBILE\_NO" value="CUSTOMER\_MOBILE\_NO">

<input type="hidden" name="EMAIL" value="CUSTOMER\_EMAIL\_ID">

<input type="hidden" name="INDUSTRY\_TYPE\_ID" value="Retail">

<input type="hidden" name="WEBSITE" value="{{ settings.PAYTM\_WEBSITE }}">

<input type="hidden" name="CHANNEL\_ID" value="WEB">

<input type="hidden" name="CALLBACK\_URL" value="{{ settings.PAYTM\_CALLBACK\_URL }}">

<button type="submit">Pay Now</button>

</form>

**callback\_page.html**:

* html

<h2>Payment Status</h2>

<p>Status: {{ status }}</p>

<p>Transaction ID: {{ txn\_id }}</p>

**8. Handling Callbacks**

Create a view to handle the payment status callback:

* python

def payment\_status(request):

status = request.GET.get('status')

txn\_id = request.GET.get('txn\_id')

# Process the payment status and transaction ID

return render(request, 'callback\_page.html', {'status': status, 'txn\_id': txn\_id})

**9. Update URLs for Callback**

Map the callback view to a URL in payment\_app/urls.py:

* python

urlpatterns = [

path('initiate\_payment/', views.initiate\_payment, name='initiate\_payment'),

path('payment\_status/', views.payment\_status, name='payment\_status'),

]