API Endpoints:

/vendors/

Description: Manages vendors.

Endpoints:

GET /vendors/: Retrieve all vendors.

POST /vendors/: Create a new vendor.

GET /vendors/{pk}/: Retrieve a specific vendor.

PUT /vendors/{pk}/: Update a specific vendor.

DELETE /vendors/{pk}/: Delete a specific vendor.

/po/

Description: Manages purchase orders.

Endpoints:

GET /po/: Retrieve all purchase orders.

POST /po/: Create a new purchase order.

GET /po/{pk}/: Retrieve a specific purchase order.

PUT /po/{pk}/: Update a specific purchase order.

DELETE /po/{pk}/: Delete a specific purchase order.

/performance/

Description: Manages historical vendor performance records.

Endpoints:

GET /performance/: Retrieve all vendor performance records.

POST /performance/: Create a new vendor performance record.

GET /performance/{pk}/: Retrieve a specific vendor performance record.

PUT /performance/{pk}/: Update a specific vendor performance record.

DELETE /performance/{pk}/: Delete a specific vendor performance record.

VendorPerformanceSerializer:

python

Copy code

class VendorPerformanceSerializer(serializers.HyperlinkedModelSerializer):

class Meta:

model = HistoricalPerformance # Assuming HistoricalPerformance is the model

fields = '\_\_all\_\_'

VendorViewSet:

python

Copy code

class VendorViewSet(viewsets.ModelViewSet):

queryset = Vendor.objects.all()

serializer\_class = VendorSerializer

lookup\_field = 'pk'

# Other configurations or methods if needed

POViewSet:

python

Copy code

class POViewSet(viewsets.ModelViewSet):

lookup\_field = 'pk'

queryset = PurchaseOrder.objects.all()

serializer\_class = POSerializer

# Other configurations or methods if needed

Routes:

python

Copy code

from django.urls import path, include

from rest\_framework import routers

from api.views import \*

router = routers.DefaultRouter()

router.register(r'vendors', VendorViewSet)

router.register(r'po', POViewSet)

router.register(r'performance', VendorPerformanceViewSet, basename='performance')

urlpatterns = [

path('', include(router.urls)),

# Other URL patterns

]