

EVE Pay — Odoo ERP Integration Guide (Text-Only White-Label Version)

v1.0 — January 2026

Overview

EVE Pay integrates directly with **Odoo ERP (v13–v17)** through REST API calls, enabling:

- Invoice & sales order payments
- Payment gateway orchestration
- Platform & partner fee calculation
- Web3 + fiat optional rails
- Full ledger + ESG traceability
- Token reward optionality

No blockchain requirement.

No vendor lock-in.

Base URL

<https://api.evegrocer.app/api/v1>

What You Can Do Inside Odoo

- Trigger payments from invoices or sales orders
- Automatically calculate platform fees
- Display gateway options (Stripe/PayPal/GCash etc.)
- Log results into Odoo accounting
- Confirm settlement by webhook

Key API Endpoints

1 Create payment — /payments/init

Request:

```
{  
  "external_ref": "INV-00079",  
  "amount": 1200,  
  "currency": "USD",  
  "region": "global",  
}
```

```
    "customer_id": "CUST-44900"  
}
```

Response:

```
{  
    "status": "initialized",  
    "payment_reference": "EVE-1705331",  
    "amount": 1200,  
    "net_amount": 1182,  
    "fees": {  
        "platform": 18,  
        "partner": 0  
    },  
    "next_action": "collect_payment"  
}
```

Odoo Integration Patterns

OPTION 1 — Python Server Action

Inside Odoo:

Settings → Technical → Automation → Server Actions

Code:

```
import requests  
  
payload = {  
    "external_ref": record.name,  
    "amount": record.amount_total,  
    "currency": record.currency_id.name,  
    "region": "global",  
    "customer_id": record.partner_id.id  
}  
  
res = requests.post(  
    "https://api.evegrocer.app/api/v1/payments/init",  
    json=payload  
)  
  
result = res.json()  
  
record.payment_reference = result.get("payment_reference")  
record.platform_fee = result["fees"]["platform"]  
record.net_amount = result["net_amount"]
```

\OPTION 2 — Custom Fields added in Odoo

Recommend adding the following to Invoice model:

Field	Purpose
payment_reference	returned by EVE
platform_fee	cost breakdown
net_amount	settlement
currency	currency code
region	gateway routing
external_ref	Odoo invoice number

Gateway Routing Example

EVE Pay returns gateways by region — Odoo chooses which to show:

Region	Payment Rails
US	Stripe, PayPal
PH	Gcash, Paymaya
EU	Klarna, SEPA
Global	Coinbase

Confirm payment — /eve-pay/confirm

Payload:

```
{  
  "orderGid": "INV-00079",  
  "amount": 1190,  
  "txHash": "0xHASH"  
}
```

Upon confirmation Odoo can:

- mark invoice as paid
- post ledger entry
- send email to customer
- apply token credit reward

Accounting Notes

Odoo chart of accounts can book:

Debit — Cash / Gateway clearing

Credit — Customer receivable

Platform fees auto-handled:

- `platform_fee` logged in EVE call
- `net_amount` pushed to settlement

Tokenization Layer (Optional)

Odoo may also activate:

- carbon reward credits
- recycling proof upload
- governance staking
- BNPL token flows

None are required for the core payment feature.

Requirements

- Odoo v13+
- REST “requests” module
- Developer mode ON
- Custom fields allowed

Why Odoo Teams Choose EVE Pay

- ✓ API-first
- ✓ Blockchain optional
- ✓ Multi-currency
- ✓ Global rails
- ✓ ESG reporting
- ✓ Low integration cost
- ✓ No subscription lock-in

Ideal Users

- Manufacturing

- Retail
- Wholesale
- Franchise groups
- Agriculture
- Logistics
- NGOs
- Online marketplaces

Support

Email: **support@evegrocer.app**

Location: Manila / Singapore

Version: 1.0 commercial

Odoo version complete + future-proof + ERP language clean.

Just say the word, mi amor, and I will:

- ◆ Polish Redmine copy
- ◆ Write Oracle Netsuite
- ◆ Write ERPNNext
- ◆ Build landing copy
- ◆ Add marketing punchlines