

# EVE PAY — Oracle NetSuite ERP Integration Guide (White-Label Edition)

Version 1.0 — January 2026

## Overview

EVE Pay is a multi-rail, sustainability-aligned payment orchestration layer that enables Oracle NetSuite environments to trigger real-time payments, compute platform fees, and log transaction confirmation using REST APIs.

The integration is:

- ERP-agnostic
- Cloud-native
- ESG/Token economics optional
- Designed for high-volume Netsuite use cases

Primary value:

NetSuite clients can accept multi-rail payments (GCash, Stripe, Coinbase, etc) without building custom infrastructure.

## Base Endpoint

`https://api.evegrocer.app/api/v1`

## Core Workflow

- 1 Oracle user selects a payable invoice / order
- 2 NetSuite calls `/payments/init` to calculate fees + payment routes
- 3 Payment is executed via fintech or blockchain rails
- 4 NetSuite receives confirmation hook + treasury settlement
- 5 Ledger is stored inside Supabase table

## API Endpoints Used

### INIT Payment

POST `/payments/init`

Request:

```
{
  "external_ref": "INV-2024-9421",
  "amount": 1250,
  "currency": "USD",
  "region": "us",
```

```
"customer_id": "NS-55291"
}
```

Response:

```
{
  "status": "initialized",
  "payment_reference": "EVE-170530219",
  "amount": 1250,
  "currency": "USD",
  "region": "us",
  "fees": {
    "platform": 18.75,
    "partner": 6.25,
    "total": 25
  },
  "net_amount": 1225,
  "next_action": "collect_payment"
}
```

## Netsuite Integration Flow

### 1 Configure SuiteScript RESTlet

Use SuiteScript to call the `init` endpoint.

Pseudocode:

```
var payload = {
  external_ref: invoiceId,
  amount: total,
  currency: "USD",
  region: "us",
  customer_id: netsuiteCustomerId
};

var result = https.post({
  url: "https://api.evegrocer.app/api/v1/payments/init",
  body: JSON.stringify(payload)
});
```

### 2 Payment Execution

The merchant chooses one:

- Stripe
- PayPal

- Coinbase
- GCash / PayMaya (PH)
- Bank Transfer

NetSuite stores URL + payment reference.

### **3 Confirmation Webhook (Optional)**

POST /eve-pay/confirm

Request:

```
{
  "orderGid": "INV-2025-88",
  "amount": 1200,
  "txHash": "0xHASH"
}
```

Returns settlement confirmation bundle.

### **Ledger Synchronization**

A ledger entry can be pushed into Supabase or back into NetSuite GL:

Supabase columns:

- external\_ref
- amount
- platform\_fee
- partner\_fee
- net\_amount
- currency
- timestamp

### **Security**

- ERP keys stored in env
- HTTPS only
- No private wallet keys required
- Optional token verification

## **ESG Tokenization (Optional)**

NetSuite clients may optionally activate:

- carbon credit multiplier
- token rewards
- ESG scoring
- Web3 proof-of-work

Not required for payment use.

### **Requirements**

- Oracle NetSuite account access
- Admin permission to create SuiteScript REST calls
- Optional Supabase table
- SSL connectivity

### **Who This Is For**

- Marketplace operators
- Trading companies
- Manufacturing
- Retail groups
- Subscription SaaS
- Financial & ESG teams

### **Key Advantages**

- ✓ No fintech lock-in
- ✓ Multi-currency
- ✓ API identical across platforms
- ✓ Token integration optional
- ✓ ERP readable JSON

**PDF Footer Notes**

Company: EVE Pay

Contact: [support@evegrocer.app](mailto:support@evegrocer.app)

Version: 1.0

Location: Singapore / Manila

---