

## BLOG 9 — A Developer’s Guide to Using the Dendrites SDK (Simple Version)

**Category:** Developer Guides

**Length:** ~800 words

---

### Developers: Safe Commerce Is Now a Few Lines of Code Away

Most blockchain integrations require:

- complex contract deployments
- writing your own escrow logic
- handling refunds manually
- managing stuck transactions
- calculating unpredictable gas
- building confirmation flows from scratch
- hacking together security checks

Dendrites changes everything.

Our SDK gives you **enterprise-grade Safe Commerce** in minutes — without writing a single line of Solidity.

If you can use JavaScript or TypeScript, you can integrate:

- **Predictable Gas™**
- **SafetySend (UNDO)**
- **APP Escrow**
- **AckPay**
- **SLA Credits**
- **PayCodes**
- **QuickPay**

into your wallet, app, marketplace, or platform.

Let’s walk you through it.

---

## 1 Installing the SDK

Setup takes a few seconds:

```
npm install @dendrites/sdk
```

or

```
yarn add @dendrites/sdk
```

Initialize your client:

```
import { Dendrites } from "@dendrites/sdk";
```

```
const dendrites = new Dendrites({  
  apiKey: process.env.DENDRITES_API_KEY,  
  network: "base-sepolia", // or mainnet later  
});
```

You now have full Safe Commerce functionality.

---

## 2 Predictable Gas™ — Get Stable Fee Quotes

Predictable Gas™ gives users **pre-quoted fee bands** with variance limits.

```
const quote = await dendrites.fees.getQuote({  
  chain: "base",  
  speed: "instant", // or "eco"  
});
```

This returns:

```
{  
  "lowerBound": "0.11 USD",  
  "upperBound": "0.19 USD",  
  "estimated": "0.15 USD"  
}
```

You can use this to cap transaction cost:

```
await dendrites.pay.send({  
  to: receiver,  
  amount: "50 USDC",  
  maxFee: quote.upperBound  
});
```

If gas spikes beyond the upper band → protocol issues **SLA Credits** automatically.

---

### **SafetySend (UNDO) — Reversible Payments**

SafetySend gives users a **3-minute undo window**.

Create a reversible transfer:

```
const tx = await dendrites.safetySend.sendWithUndo({  
  to: receiverAddress,  
  amount: "25 USDC",  
  undoWindow: 180 // seconds  
});
```

If the user wants to cancel:

```
await dendrites.safetySend.undo(tx.id);
```

If not undone → funds finalize automatically.

This prevents accidental transfers, scam sends, wrong-chain mistakes, and phishing damage.

---

### **AckPay — Receiver Must Accept Payment**

AckPay turns payments into **pending settlements**.

**Sender side:**

```
const request = await dendrites.ackPay.create({
```

```
to: merchantAddress,  
amount: "100 USDC",  
timeout: 300  
});
```

#### **Receiver side:**

```
await dendrites.ackPay.accept(request.id);
```

#### **Auto-refund if not accepted:**

```
await dendrites.ackPay.autoRefund(request.id);
```

This protects buyers and forces merchants to acknowledge the payment.

---

## **5 APP Escrow — Milestones, Delivery, Approvals, Refunds**

APP Escrow is the first true **programmable on-chain workflow system**.

Create an escrow:

```
const escrow = await dendrites.escrow.create({  
  payer: sender,  
  payee: freelancer,  
  amount: "200 USDC",  
  milestones: [  
    { id: "design", amount: "60 USDC" },  
    { id: "frontend", amount: "70 USDC" },  
    { id: "backend", amount: "70 USDC" }  
  ],  
  timeout: 86400  
});
```

Release milestone:

```
await dendrites.escrow.release({
```

```
escrowId: escrow.id,  
milestoneId: "design"  
});
```

Refund automatically if deadlines fail:

```
await dendrites.escrow.refund(escrow.id);
```

This enables:

- gig platforms
  - agency payments
  - multi-step services
  - e-commerce protection
  - conditional workflow payments
- 

## 6 QuickPay — Universal Payment Requests

QuickPay lets merchants generate a **single link or QR** that includes:

- amount
- asset
- chain
- fee tier
- optional escrow/UNDO/AckPay settings
- metadata

```
const paycode = await dendrites.paycodes.generate({  
  amount: "15 USDC",  
  asset: "USDC",  
  chain: "base",  
  features: {  
    undo: true,
```

```
    ack: true,  
    escrow: false  
  }  
});
```

Output example:

```
{  
  "paycode": "dndx:base:pay:0xabc123?...",  
  "qr": "data:image/png;base64,...",  
  "link": "https://pay.dendrites.ai/..."  
}
```

PayCodes behave like **EIP-681**, but enhanced for Safe Commerce.

---

## 7 Testnet Instructions

Dendrites testnet runs on Base Sepolia during presale.

To start testing:

```
export DENDRITES_API_KEY="test_api_key_here"
```

Fund your wallet with Base Sepolia ETH and USDC (faucets provided on the testnet dashboard).

Run any of the flows above — everything works exactly like mainnet, but with simulated Safe Commerce logic.

---

## 8 Why Developers Love the Dendrites SDK

### ✓ No Solidity needed

Everything is API + SDK based.

### ✓ Safe Commerce defaults

Every transaction is structured, predictable, and protected.

### **✓ Production-ready workflows**

Use the same logic used by gig platforms, marketplaces, and payment networks.

### **✓ Predictable fees**

No more gas chaos.

### **✓ Easy integration**

Wallets, marketplaces, remittance apps — everything plugs in instantly.

---

## **Closing: Web3 Payments Can Finally Match Web2 Quality**

With the Dendrites SDK, developers can give users:

- reversible payments
- predictable fees
- milestone escrow
- confirmation-based settlement
- auto-refunds
- fraud protection
- enterprise workflows

in **under 10 minutes**.

This is how digital money becomes truly usable — not just powerful.

Dendrites gives you the tools.

You build the future.