Razorpay Payment & Refund Flow

1. Introduction

This document explains how Razorpay handles payments and refunds, covering frontend checkout, backend verification, settlement, and refund handling (including per-product refunds).

2. Razorpay Payment Flow

Step 1: Creating an Order (Backend → Razorpay)

• The backend calls Razorpay's Orders API to create an order with total amount, currency, and receipt ID. • Razorpay generates an `order_id` (e.g., order_xxxx) and returns it. • This `order_id` is sent back to the frontend. ■ At this stage, no money is deducted from the user — only an order record is created.

Step 2: Frontend Checkout

• The frontend loads Razorpay Checkout with: – key_id (public key) – amount – order_id – user details (name, email, contact) • Razorpay Checkout popup opens (UI provided by Razorpay). • User enters payment method (card/UPI/netbanking) and authorizes payment. ■ Money moves from customer's account → Razorpay escrow.

Step 3: Payment Success Callback

• Razorpay sends payment details back to frontend: – razorpay_order_id – razorpay_payment_id – razorpay_signature • At this point: – Money is deducted from customer's bank – Your backend still needs to verify signature.

Step 4: Backend Payment Verification

• Backend regenerates HMAC signature using `razorpay_order_id|razorpay_payment_id` and secret key. • If signature matches → Payment is genuine. • If not → Possible fraud (reject transaction).

Step 5: Order Confirmation

• After verification, backend: – Saves order in MongoDB – Updates user's total orders & spends – Sends confirmation email.

Step 6: Settlement

• Razorpay holds amount in escrow for some hours/days. • Then transfers to your merchant bank account (after deducting fees). • Backend is not involved in settlement.

3. Security in Razorpay Checkout

• Key ID (Public) → Safe to share with frontend. • Key Secret (Private) → Must remain in backend. Used for: – Creating Orders – Verifying signatures – Issuing refunds. ■ Security

Role: Prevents fake success callbacks and ensures payments cannot be forged.

4. Razorpay Refund Flow

Step 1: Where the Money Is After Payment

• Money is first stored in Razorpay escrow (not merchant bank). • Refunds are possible because Razorpay holds this amount temporarily.

Step 2: How Refund Works

• Merchant backend calls Razorpay Refund API with `paymentId` and amount. • Razorpay validates payment & creates refund request. • Refund is initiated through original payment rails (UPI, card network, or netbanking).

Step 3: Refund Processing Timelines

• UPI → Minutes to hours • Cards → 5–7 business days • NetBanking → 2–5 business days ■ Refund always goes to the same payment source.

Step 4: Settlement vs Refund

• If money still in escrow → Razorpay refunds directly. • If already settled → Razorpay refunds from its pool and deducts in your future settlements.

5. Per-Product Refund Handling

• Your system creates a single Razorpay Order for the entire cart. • Problem: If one product is cancelled, you need partial refunds. • Solution: Razorpay supports partial refunds using the same paymentId. • Refund amount = cancelled product price. • Update DB to mark only that product order as cancelled and refund initiated.

Sample Cancel Product Controller

1. Find product order by ID 2. Call Razorpay Refund API with that product’s price 3. Update DB orderStatus = 'cancelled', paymentInfo = 'refunded' 4. Adjust user stats (totalOrders, totalSpends) 5. Notify user via email