# Automating Escrow Payouts on Stripe Connect for Orla3

This document outlines how to convert the current manual payout process into a fully automated escrow release and refund system using Stripe Connect. It is designed for implementation by the Orla3 engineering team.

## 1. Use the Right Connect Mode

Use 'Separate Charges and Transfers' mode. Buyer funds go to the platform account first, allowing the platform to hold them (escrow-like) until the buyer accepts delivery.

## 2. Capture Buyer Payment

Create a PaymentIntent on the platform account. On success, store charge.id, order\_id, connected\_account\_id, and net\_to\_creator (after fees). This marks the job as funded.

## 3. Webhooks

Implement a webhook listener for payment\_intent.succeeded or charge.succeeded to mark the job as funded and persist charge.id. Verify Stripe webhook signatures.

## 4. Automate Release on Acceptance

When buyer downloads or presses 'Approve', trigger a Transfer API call:

POST /v1/transfers  
{  
 amount: net\_to\_creator,  
 currency,  
 destination: connected\_account\_id,  
 source\_transaction: charge.id  
}

This moves funds from the platform balance to the seller’s connected account. The remaining balance (your fee) remains as platform revenue.

## 5. Automate Payouts

Option 1: Enable automatic payout schedules for sellers. Option 2: Keep manual but create payouts programmatically when a transfer is complete.

## 6. Automate Refunds

If buyer rejects before transfer: create a refund on the original charge.  
If transfer occurred: create a transfer reversal, then refund the buyer.

## 7. Add Safety Timer

Add a scheduled job to auto-release funds after a set period (e.g., 7 days of no buyer response). Define this SLA in terms and conditions.

## 8. Compliance Checks

Ensure each seller’s connected account has KYC verified and payouts\_enabled=true. Ensure a valid bank account or card is attached.

## 9. Reliability Practices

Use idempotency keys for all transfer/refund/payout requests. Log all Stripe API calls with request/response and your order\_id for audit.

## Example Code (Node.js)

// Charge buyer  
const pi = await stripe.paymentIntents.create({  
 amount, currency,  
 automatic\_payment\_methods: { enabled: true },  
 metadata: { order\_id, seller: connectedAccountId }  
});  
  
// Release to creator  
await stripe.transfers.create({  
 amount: netToCreator,  
 currency,  
 destination: connectedAccountId,  
 source\_transaction: chargeId,  
 metadata: { order\_id }  
});  
  
// Refund buyer  
await stripe.refunds.create({  
 charge: chargeId,  
 amount: refundAmount  
});

## Pitfalls to Avoid

• Avoid Destination Charges if you need escrow.  
• Always set source\_transaction on transfers.  
• Make webhook actions idempotent.  
• Verify all Stripe signatures.

This architecture ensures Orla3 maintains escrow control, automatically releases or refunds based on buyer actions, and recognizes only platform fees as revenue.