Payment Flow Explanation

1. **Key Steps**
    - **Checkout & Order Creation:** When user initiates checkout, merchant creates an order record in merchant database with a PENDING state. This creates an order record with order ID before communicating with payment gateway.
    - **Payment Intent:** Merchant backend requests a "Payment Intent" from payment gateway sending payment details i.e. amount, currency, etc. Also, idempotency key is created at this step from merchant backend, which can be an order ID. Then payment gateway creates a transaction ID and returns redirect URL of hosted payment page or token for embedded page.
    - **Redirect/Embedded Payment:** Merchant redirect user to payment gateway's hosted page (or serve an embedded page i.e. iframe). This ensures PCI data (PAN/CVV) goes directly to payment gateway, bypassing merchant system which might not PCI-DSS compliance.
    - **Webhook & Update:** Once the payment is processed, payment gateway sends a webhook (POST request) to merchant. Merchant consumes the event and update order state from PENDING to PAID, FAILED, or EXPIRED in merchant DB.
2. **Handling Failures & Reliability**
    - **Webhook Retry Strategy:** If merchant fails to acknowledge a webhook with OK status (due to downtime or timeout etc.), payment gateway triggers an exponential backoff retry policy (e.g., retrying at 1 minute, 5 minutes, 1 hour). If retries exhausted, payment gateway place event in a dead letter queue (DLQ) with an alert for manual reconciliation.
    - **Idempotency Key Usage:** To prevent duplicate charges during network flakiness, merchant must generate and include a unique Idempotency-Key (can be order ID or a UUID). If payment gateway receives the same key twice, it returns the cached result of the original successful request rather than processing a new charge.
    - **Timeout / Expiry Rules:** Merchant implement a TTL (time to live) of 15–30 minutes for PENDING order. A background worker periodically checks for PENDING orders older than this threshold and marks them EXPIRED after a final check with payment gateway.
    - **Reconciliation:** Merchant creates a scheduled job fetches a daily settlement report from payment gateway and compares it against merchant database to flag and resolve status mismatches.
3. **Security Considerations**
    - **TLS & Encryption:** All requests happen over HTTPS. API keys and webhook secrets are injected via environment variables or a secrets manager i.e. HashiCorp Vault.
    - **No Sensitive Data Logging:** Never log, store, or print PAN or CVV. Logs should only contain non-sensitive references like transaction ID, order ID, or masked data.
    - **PII Privacy:** PII data is stored with access controls. Only necessary services have access to read user contact details. May utilize OPA (Open Policy Agent) for policy management.
    - **Signature Verification:** Merchant must verify the signature included in the webhook headers using shared secret, ensure that the request comes from payment gateway.
4. **What to Monitor**
    - **Payment Success Rate (SR):** Successful Payments / Total Attempts (metrics). Alert when SR decreases, as this can indicate issues. Alternatively, monitor the number of error logs and set thresholds to alert when log count exceed.
    - **Webhook Latency & Failures:** Monitor the P99 webhooks processing time and set alert for spikes in non-200 responses.