Transaction Flow

Step 1: Enabling Production API Key

Before initiating a transaction, the merchant must have their **Production API Key** enabled. This can be done through the **LightSpeedPay Admin** panel. Once the key is enabled, the merchant can use the API to start processing transactions.

Step 2: Merchant Server Whitelisting

To ensure security, the merchant's server (from where they will initiate transaction requests) must be **whitelisted** by LightSpeedPay. This ensures that only authorized requests are allowed to interact with the LightSpeedPay API.

Step 3: Initiating the Transaction

After the merchant's server is whitelisted and the production API key is enabled, the merchant can initiate a payment request by calling the **Initiate Request API** from their server.

Upon successfully initiating a transaction, the LightSpeedPay system will return a unique `trxnId` to the merchant. The merchant will then use this transaction ID (`trxnId`) along with their specific API key and secret to proceed with the payment process.

- **API Endpoint for Initiating a Request**: `[Add Your API Endpoint Here]`
- **API Response Example**:

```
{
  "status": "success",
  "trxnId": "66e59f687b0b91cce91852b5"
}
```

Step 4: Redirecting to the Payment Page

After receiving the `trxnId`, the merchant must redirect the user to the payment page hosted by LightSpeedPay. The URL format is as follows:

```
https://pay.lightspeedpay.in/?txn={transaction_id}&key={api_key}&secret=
{api_secret}
```

- Example:

  ```
  https://pay.lightspeedpay.in/?
  txn=66e59f687b0b91cce91852b5&key=6f1b5113-3ab6-4222-ace5-
  de0d2d09fade&secret=krc5jVdh2rCkMIdzH4LgerH1XP28CAei
  ```

Here:

- `txn` refers to the `trxnId` received during the initiation step.
- `key` refers to the **API key** for the merchant.
- `secret` refers to the **API secret** for the merchant.

Step 5: QR Code Generation and Frontend Processing

When the user lands on the payment page, the system internally calls the **Request QR Code API**, which generates a **QR Intent**. This QR can be scanned using a mobile device, or the payment can be processed directly from a desktop device.

The frontend will handle the display of the QR code and any related user interface elements. It will also handle the logic for checking the status of the transaction.

Step 6: Polling for Transaction Status

Once the payment page is displayed, it will continuously poll the backend to check the status of the transaction. The polling will occur every **3 seconds** (this interval can be configured on the frontend) for a duration of **10 minutes**.

- During this time, the system checks if the transaction has either **succeeded** or **failed**.
- The polling can be implemented using an AJAX call or WebSockets to check the transaction status every 3 seconds.

Step 7: Transaction Status and Redirection

Once the transaction is completed (either **failed** or **successful**), the user will be redirected to the respective URL provided by the merchant.

- **Success URL**: The URL where the user will be redirected if the payment is successful.
- **Failure URL**: The URL where the user will be redirected if the payment fails.

The merchant can configure these URLs during the initiation of the transaction.

Example Transaction Flow

1. **Merchant API Call**:
    - Merchant hits the initiate request API from their whitelisted server.
    - LightSpeedPay returns a unique `trxnId`.
2. **Redirect to Payment Page**:
    - Merchant redirects the user to the payment page with `txn`, `key`, and `secret` in the URL.
3. **Frontend Processes the Transaction**:
    - The payment page generates the QR code and continuously polls the backend for transaction status.
    - Polling occurs every 3 seconds for up to 10 minutes.

4. **Transaction Completion**:
    - Upon success or failure, the user is redirected to the appropriate URL.

This section outlines the full flow, providing merchants with a clear understanding of how transactions are initiated and processed within the LightSpeedPay system.