Explicación de Integración API Khipu - Consola y Tkinter GUI

Este documento describe el desarrollo y justificación técnica de dos scripts que integran la API de pagos de Khipu en modo desarrollador. Ambas versiones comparten la misma lógica, diferenciándose solo en el método de interacción con el usuario: uno por consola y otro mediante interfaz gráfica Tkinter.

# Paso a Paso y Referencias Documentales

## 1. Autenticación con API Key

Ambos scripts utilizan autenticación vía API Key, según la documentación oficial de Khipu API 3.0. Esto se realiza enviando el valor de la clave secreta mediante la cabecera *x-api-key.*

Referencia: <https://docs.khipu.com/portal/es/payment-auth/>

## 2. Endpoint utilizado

Se utiliza el endpoint POST *https://payment-api.khipu.com/v3/payments*, tal como lo indica la sección *Crear un cobro* de la documentación. Este permite generar una transacción de pago.

Referencia: [https://docs.khipu.com/portal/es/payment-api/#crear-un-cobro](https://docs.khipu.com/portal/es/payment-api/%23crear-un-cobro)

## 3. Parámetros enviados en la solicitud

Los campos enviados en el cuerpo del POST son:

* *amount:* Monto del pago
* *currency:* 'CLP' según entorno chileno
* *subject:* Descripción del pago
* *payer\_email:* Correo del cliente (opcional pero incluido para trazabilidad)
* *transaction\_id:* ID único de la transacción (recomendado para evitar duplicados)
* *return\_url y cancel\_url:* URLs de redirección del flujo

Estos parámetros están detallados en la especificación del método POST /payments.

Referencia: [https://docs.khipu.com/portal/es/payment-api/#crear-un-cobro](https://docs.khipu.com/portal/es/payment-api/%23crear-un-cobro)

## 4. Validaciones de entrada

Se implementaron validaciones para asegurar datos correctos:

* Monto debe ser un número entero positivo
* Descripción no puede estar vacía
* Correo debe tener formato válido (expresión regular)
* Se permite cancelar la operación escribiendo 'cancelar' (en consola) o cerrando la ventana (GUI)

Inspirado en buenas prácticas de validación y control de errores de la documentación general de integración.

## 5. Flujo de ejecución en ambas versiones

* En consola: Se guía al usuario paso a paso usando inputs, prints con emoticones y muestra del resultado final con link.
* En GUI (Tkinter): Se presenta una ventana con tres campos (monto, descripción, correo) y botón *'Crear Pago'*.
* En ambos casos, se abre el navegador automáticamente con la URL de pago (*payment\_url*) obtenida desde la respuesta de Khipu.

Referencia: Campo *payment\_url* entregado en la respuesta del método POST /payments.

## 6. Modo desarrollador y pruebas

La integración se realizó con una cuenta en modo desarrollador, usando DemoBank como entorno de pruebas. Esto es clave para realizar pagos sin transacciones reales.

Referencia[: https://docs.khipu.com/portal/es/payment-api/#ambiente-de-pruebas-y-paso-a-produccion](:%20https:/docs.khipu.com/portal/es/payment-api/%23ambiente-de-pruebas-y-paso-a-produccion)

## 7. Verificación del estado de un pago

Además de generar pagos, se probó la funcionalidad de consulta de estado mediante el endpoint GET *https://payment-api.khipu.com/v3/payments/{payment\_id}.* Esta funcionalidad permite verificar manualmente si un pago fue conciliado exitosamente.

En la respuesta obtenida, los campos *"status": "done"* y *"conciliation\_date"* confirmaron que el pago fue ejecutado exitosamente. Estos son los indicadores clave que validan la conciliación final en la API.

Referencia: <https://docs.khipu.com/openapi/es/v1/instantpayment/openapi/operation/getPaymentById/>

# Conclusión

Ambos scripts son funcionales, robustos y comprensibles. Se desarrollaron basándose directamente en la documentación oficial de Khipu y aplican correctamente los conceptos técnicos esperados. Esta solución demuestra autonomía, comprensión técnica y capacidad de aplicar documentación para resolver desafíos reales.

Demostración Visual - Integración API Khipu

Este documento presenta capturas simuladas del funcionamiento de ambos scripts desarrollados para integrar la API de pagos de Khipu. Se muestran la versión por consola y la versión con interfaz gráfica usando Tkinter.

# Versión por Consola

Simulación de la ejecución desde terminal, mostrando el ingreso de datos, validación y respuesta de la API:

Texto

El contenido generado por IA puede ser incorrecto.

Interfaz de usuario gráfica, Aplicación

El contenido generado por IA puede ser incorrecto.

Interfaz de usuario gráfica, Aplicación

El contenido generado por IA puede ser incorrecto.Interfaz de usuario gráfica, Aplicación

El contenido generado por IA puede ser incorrecto.

Interfaz de usuario gráfica, Aplicación

El contenido generado por IA puede ser incorrecto.Interfaz de usuario gráfica, Aplicación

El contenido generado por IA puede ser incorrecto.Interfaz de usuario gráfica, Aplicación

El contenido generado por IA puede ser incorrecto.

# Versión con Interfaz Gráfica (Tkinter)

Simulación de la ventana interactiva con campos de entrada y respuesta visual tras crear el pago:

Interfaz de usuario gráfica, Aplicación

El contenido generado por IA puede ser incorrecto.

Interfaz de usuario gráfica, Aplicación

El contenido generado por IA puede ser incorrecto.Interfaz de usuario gráfica, Aplicación

El contenido generado por IA puede ser incorrecto.Interfaz de usuario gráfica, Aplicación

El contenido generado por IA puede ser incorrecto.Interfaz de usuario gráfica, Aplicación

El contenido generado por IA puede ser incorrecto.Interfaz de usuario gráfica, Aplicación

El contenido generado por IA puede ser incorrecto.Interfaz de usuario gráfica, Aplicación

El contenido generado por IA puede ser incorrecto.

**Verificación del estado de un pago**

Para ello, se utilizó Postman, autenticando con *x-api-key*, y se comprobó que el *payment\_id* debe ir sin guiones (ej: drp7wjy8qfoq). El resultado fue exitoso, devolviendo información como el estado *"done",* monto, correo del pagador y enlace al comprobante (*receipt\_url*).

Una captura de pantalla de una red social

El contenido generado por IA puede ser incorrecto.

{

    "amount": "3.0000",

    "app\_url": "khipu:///pos/drp7wjy8qfoq",

    "attachment\_urls": [],

    "bank": "DemoBank",

    "bank\_account\_number": "969279012013",

    "bank\_id": "Bawdf",

    "body": "",

    "cancel\_url": "https://www.khipu.com/",

    "conciliation\_date": "2025-05-09T18:20:28.884Z",

    "currency": "CLP",

    "custom": "",

    "discount": "0.0000",

    "expires\_date": "2025-05-10T18:19:29.309Z",

    "funds\_source": "not-available",

    "notification\_token": "3d6c1abc2f545e64fe9508f54330da65eabdc63ce405bcbced0c26e6d05a7de0",

    "notify\_api\_version": "",

    "notify\_url": "",

    "out\_of\_date\_conciliation": false,

    "payer\_email": "dkn.ddr@gmail.com",

    "payer\_name": "Cobrador de desarrollo #497.684",

    "payment\_id": "drp7wjy8qfoq",

    "payment\_method": "simplified\_transfer",

    "payment\_url": "https://khipu.com/payment/info/drp7wjy8qfoq",

    "personal\_identifier": "16.142.874-4",

    "picture\_url": "",

    "ready\_for\_terminal": true,

    "receipt\_url": "https://s3.amazonaws.com/notifications.khipu.com/CPKH-0905251420-drp7wjy8qfoq.pdf",

    "receiver\_id": 497684,

    "responsible\_user\_email": "dkn.ddr@gmail.com",

    "return\_url": "https://www.khipu.com/",

    "send\_email": false,

    "send\_reminders": false,

    "simplified\_transfer\_url": "https://app.khipu.com/payment/simplified/drp7wjy8qfoq",

    "status": "done",

    "status\_detail": "normal",

    "subject": "3",

    "third\_party\_authorization\_details": "",

    "transaction\_id": "tx-20250509141928",

    "transfer\_url": "https://khipu.com/payment/manual/drp7wjy8qfoq"

}

# Resumen

Ambas versiones permiten crear pagos reales en modo desarrollador utilizando DemoBank. La versión por consola es ideal para uso técnico, mientras que la GUI en Tkinter ofrece una experiencia más visual para usuarios finales.