



KIA PAY

Developer & Business Integration Guide

Official Integration Documentation • Version 1.0 • September 2026

Developer API • Business Portals • WordPress / Elementor
Webhooks • Receipts • Payment Integration

pay.kiahelps.com



CONTENTS

Click any section to jump directly to it.

- [1. What Is the KIA-PAY API?](#)
- [2. What Does the KIA-PAY API Do?](#)
- [3. How Does It Work?](#)
- [4. Where Can the KIA-PAY API Be Used?](#)
- [5. Integration Models](#)
- [6. Production API Endpoints](#)
- [7. Authentication & API Keys](#)
- [8. Create Payment - Supported Input](#)
- [9. Example Create Payment Request](#)
- [10. Customer Information, ID Privacy & Tax](#)
- [11. Payment Status & Expiration](#)
- [12. Webhooks - Automatic Business Updates](#)
- [13. Receipts & Verification](#)
- [14. WordPress & Elementor Widget](#)
- [15. Setting Up the Widget in a Business Portal](#)
- [16. Integrating KIA-PAY into a Custom Business Portal](#)
- [17. Security Requirements](#)
- [18. WordPress Public Distribution](#)
- [19. Current Integration Checkpoint](#)
- [20. Planned Documentation Expansion](#)



1. What Is the KIA-PAY API?

The KIA-PAY API is the secure integration layer that allows a business website, customer portal, e-commerce system, invoicing platform, POS, CRM/ERP, SaaS application, mobile application, or other business software to create and manage KIA-PAY payment requests without giving that external system the merchant's KIA-PAY username and password.

Instead, the business uses its own Developer API key on a trusted server. The business system sends payment information to KIA-PAY, KIA-PAY creates the payment request and hosted payment page, the customer completes the payment, KIA-PAY verifies completion, and the business can receive the result through the payment-status API and signed webhook notifications.

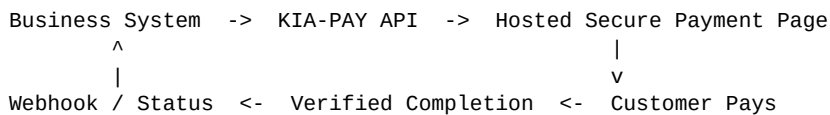
2. What Does the KIA-PAY API Do?

The API can connect KIA-PAY payment capability to an existing business workflow rather than forcing the business to rebuild its portal around KIA-PAY.

- Create a payment request from a business website, portal, application, invoice, order, or internal system.
- Send amount, currency, merchant reference, customer information, notes, and merchant-defined tax information to KIA-PAY.
- Return a KIA-PAY payment ID and hosted payment URL that can be presented to the customer.
- Allow KIA-PAY to handle the customer-facing payment experience and payment verification.
- Allow the business backend to retrieve payment status.
- Notify the business automatically when a verified payment is completed through a signed webhook.
- Support receipt generation, receipt email workflows, verification, and reconciliation after completion.

3. How Does It Work?

At a high level, the integration follows this flow:



The rest of this guide explains the HOW: authentication, API keys, payment creation, customer and tax fields, payment status, webhooks, receipts, the WordPress/Elementor widget, business-portal integration, and security.

4. Where Can the KIA-PAY API Be Used?

Integration Type	How KIA-PAY Can Be Used
Business & Customer Portals	Create payment requests for invoices, accounts, subscriptions, services, reservations, or orders inside an existing portal.
E-commerce Websites	Create a KIA-PAY payment for an online order and update the business order after verified completion.
WordPress & Elementor	Use the KIA-PAY WordPress plugin and Elementor integration without exposing the API key in the browser.
WooCommerce	Planned gateway integration can use the same API/webhook architecture to create checkout payments and update WooCommerce order status.
Custom Websites & Web Apps	Integrate from server-side applications such as Next.js/Node.js, PHP/Laravel, Python/Django, ASP.NET, Java, or other backend frameworks.
POS & Invoicing Systems	Generate payment requests tied to an invoice/reference and present or send the KIA-PAY payment URL to the customer.
Accounting, ERP & CRM	Associate KIA-PAY payment IDs and merchant



	references with internal customer/invoice records and reconcile completed payments.
SaaS Platforms	Add KIA-PAY as a payment capability inside a hosted business application while keeping API credentials server-side.
Mobile Applications	Use the application's trusted backend to create payments, then open or share the returned hosted payment URL.
Remote Payments	Create a payment remotely and deliver the hosted payment link through the business's own email, SMS, portal, or messaging workflow.
In-Person Assisted Payments	Create a payment request at a business location and present the KIA-PAY payment experience to the customer.
Automation & Fulfillment	Use verified payment.completed webhooks to mark an invoice paid, complete an order, activate a service, or trigger another authorized business workflow.

5. Integration Models

KIA-PAY can be integrated at different levels depending on the business's technical environment.

- No-code / low-code: WordPress plugin + Elementor widget or shortcode.
- Business portal integration: the portal backend calls the Developer API and uses webhooks/status to synchronize payment state.
- Custom application integration: developers call the API directly from their trusted server-side application.
- Commerce integration: a checkout or order system creates payments and uses verified completion to update order state.

6. Production API Endpoints

Purpose	Method	Endpoint
Health check	GET	https://pay.kiahelps.com/api/v1/health
Create payment	POST	https://pay.kiahelps.com/api/v1/payments
Payment status	GET	https://pay.kiahelps.com/api/v1/payments/{paymentId}

7. Authentication & API Keys

External integrations authenticate with a merchant-owned Developer API key. Send the key as a Bearer token in the Authorization header. Store it only on the trusted business server. Never place it in browser JavaScript, public HTML, Elementor custom code, mobile client source, or a public repository.

Use an Idempotency-Key when creating a payment so an application retry does not unintentionally create a duplicate payment.

8. Create Payment - Supported Input

Field	Type	Required	Description
amount	number/string	Yes	Payment amount before tax.
currency	string	Yes	Payment currency supported by KIA-PAY.
merchant_reference	string	No	Business invoice, order, or reference identifier.
customer_note	string	No	Note associated with the customer/payment context.



merchant_note	string	No	Merchant-side note associated with the payment.
customer_name	string	No	Customer full name; can prefill payer information.
customer_email	string	No	Customer email; can prefill payer information and support receipt delivery.
customer_phone	string	No	Customer phone number.
customer_address	string	No	Customer address.
customer_id_type	string	No	drivers_license, passport, national_id, or other.
customer_id_number	string	No	Customer ID input; only the last 4 characters are persisted in the payment record.
apply_tax	boolean	No	Whether tax should be applied.
tax_name	string	No	Merchant-defined tax label such as Sales Tax, VAT, GST, or HST.
tax_rate	number/string	No	Merchant-defined tax percentage rate.

9. Example Create Payment Request

```
POST /api/v1/payments
Authorization: Bearer YOUR_API_KEY
Content-Type: application/json
Idempotency-Key: order-1001
```

```
{
  "amount": 100,
  "currency": "USD",
  "merchant_reference": "ORDER-1001",
  "customer_name": "Example Customer",
  "customer_email": "customer@example.com",
  "customer_phone": "555-123-4567",
  "customer_address": "123 Example Street",
  "customer_id_type": "drivers_license",
  "customer_id_number": "EXAMPLE1234",
  "customer_note": "Payment for Order 1001",
  "merchant_note": "Online order",
  "apply_tax": true,
  "tax_name": "Sales Tax",
  "tax_rate": "5"
}
```

A successful create operation returns the KIA-PAY payment information, including a payment identifier and hosted payment_url. The business should redirect or link the customer to that hosted KIA-PAY payment page.

10. Customer Information, ID Privacy & Tax

Customer information supplied by the business can prefill the payer-information section of the hosted payment flow. For customer_id_number, the payment record is designed to persist only payer_id_last4 together with the ID type; the full ID number is not persisted in the payment record.

Tax names and rates are merchant-defined. KIA-PAY calculates the tax amount from the submitted amount and rate and shows the breakdown in the relevant payment and receipt views. The business remains responsible for choosing the appropriate tax type and rate.



11. Payment Status & Expiration

Use GET `/api/v1/payments/{paymentId}` from the business backend to retrieve payment status. Current Developer API and Merchant-created payment requests use a 24-hour expiry window. A payment should be treated as paid only after KIA-PAY's verification flow has marked it Completed.

12. Webhooks - Automatic Business Updates

KIA-PAY supports payment.completed webhook delivery for external integrations. The business configures a webhook destination and verifies signed delivery. Webhook processing should be idempotent and should validate the signature, timestamp/event identity, and replay state before changing a local order or invoice.

Typical automation after verified completion:

- Mark an invoice as Paid.
- Move an e-commerce order to its paid/processing state.
- Activate an authorized service or account entitlement.
- Record the KIA-PAY payment ID/transaction reference for reconciliation.
- Trigger the business's own confirmation or fulfillment workflow.

13. Receipts & Verification

Completed payments support receipt data, receipt hashes, and automatic receipt-email workflows when an eligible customer email is available. Receipt hashes are generated for completed payments, not for pending, expired, cancelled, or failed payments.

14. WordPress & Elementor Widget

The KIA-PAY WordPress plugin provides a server-side bridge between a WordPress website and the KIA-PAY Developer API. The Developer API key is saved in the WordPress plugin settings and is used by the WordPress server; it is not meant to be exposed to visitors.

The integrated portal shortcode is:

```
[kia_pay_portal]
```

It renders the KIA-PAY payment-creation interface together with the integrated business-dashboard access section. The plugin also supports a standalone payment widget/shortcode and an Elementor widget for visual page building.

15. Setting Up the Widget in a Business Portal

A WordPress business can use the following setup sequence:

1. Install and activate the KIA-PAY WordPress plugin.
2. Create a Developer API key for the business in KIA-PAY and store it in the plugin's KIA-PAY Connection settings.
3. Confirm the KIA-PAY API base URL is set to the production KIA-PAY service.
4. Create or edit the desired WordPress/Elementor page.
5. Add the KIA-PAY Elementor widget, the standalone payment shortcode, or `[kia_pay_portal]` for the integrated portal experience.
6. Publish the page and test payment creation from the public website.
7. Confirm the created payment opens on the hosted KIA-PAY payment page and is associated with the correct merchant.
8. Complete a controlled test payment and confirm status, receipt, and business-side visibility.



16. Integrating KIA-PAY into a Custom Business Portal

For a custom portal, the customer-facing browser should call the business's own backend. The business backend then calls KIA-PAY with the private Developer API key. This prevents the API key from being exposed to the customer.

```
Customer Browser
-> Business Portal Backend
   -> KIA-PAY Developer API
       -> payment_id + payment_url
<- Business Portal Backend
<- Redirect / Open KIA-PAY payment_url
```

After payment, the business can use the signed payment.completed webhook as the primary automatic update mechanism and the payment-status endpoint for server-side status checks or reconciliation.

17. Security Requirements

- Keep Developer API keys and webhook secrets server-side.
- Use a separate API key for each intended merchant/integration environment where appropriate.
- Revoke and replace a key if it is exposed.
- Use Idempotency-Key for create-payment retries.
- Validate webhook signatures and protect against replay/duplicate processing.
- Do not authorize a payment or business action only from merchant_reference.
- Do not log plaintext API keys, webhook secrets, or full customer ID numbers.
- For fixed-price commerce, derive or verify the amount on the trusted business server rather than trusting browser-submitted values.
- Treat KIA-PAY verified Completed status as the payment authority.

18. WordPress Public Distribution

The KIATOKEN KIA-PAY WordPress/Elementor plugin version 1.0.0 has completed the project's release preparation and Plugin Check validation and has been submitted to the WordPress.org Plugin Directory. The submission is currently Awaiting Review by the WordPress.org review team. Public directory availability will begin after WordPress.org approval.

19. WordPress Integration Status

Plugin: KIATOKEN KIA-PAY

Version: 1.0.0

WordPress.org status: Submitted - Awaiting Review

Integration support: WordPress shortcode, integrated portal, and Elementor widget.

20. Planned Documentation Expansion

Future revisions of this guide will be expanded with exact response schemas, error/status-code examples, webhook signature/header examples, API-key management screenshots, WordPress installation screenshots, Elementor configuration screenshots, troubleshooting, and WooCommerce documentation after that gateway is implemented.

KIA-PAY | pay.kiahelps.com