

PAYSTACK SUBSCRIPTION AND PAYMENT MANAGEMENT SYSTEM

This project is a simple web-based system that connects with Paystack to help you manage payments, subscriptions, and plans. You can use it to test how Paystack subscriptions work and to view all related transactions easily.

WHAT THE SYSTEM DOES

1. PLANS

Plans are like payment packages that users can subscribe to.

For example:

- Basic Plan – ₦1000 per month
- Premium Plan – ₦3000 per month
- VIP Plan – ₦5000 per month

You can create a plan and list all the existing ones on Paystack.

Each plan has a unique code (called `plan_code`) which Paystack uses to identify it.

2. SUBSCRIPTIONS

A subscription links a customer to a specific plan. Once subscribed, Paystack automatically charges the customer based on the plan's interval (e.g. monthly).

For example:

- If "John" subscribes to the "Premium Plan", Paystack will charge John ₦3000 monthly.

You can:

- Subscribe a customer to a plan.
- Disable (cancel) a customer's subscription.
- List all subscriptions in a simple table.

3. TRANSACTIONS

Transactions are all the payments that have been made by customers.

You can see:

- Who made the payment
- The amount
- The date
- The status (successful or failed)

The list supports pagination, meaning you can view them page by page.

Example:

- Page 1 → first 10 transactions
- Page 2 → next 10 transactions

4. INSTALLMENT PAYMENTS

Installment payments allow customers to pay large amounts in smaller monthly chunks.

For example:

- A customer wants to pay ₦100,000 but can't afford it all at once.
- They can split it into ₦10,000 monthly payments over 10 months.

You can:

- Create an installment plan for a customer.
- View all active installments.
- See payment history for each installment.
- The system can automatically charge customers when their payment is due.

5. WEBHOOKS

Paystack sends automatic notifications (called webhooks) to your server whenever something happens — for example, when a user pays successfully or when a subscription fails.

This means you don't have to manually check; your app is automatically informed.

Example of webhook events:

- `charge.success` → when a payment succeeds.
- `invoice.payment_failed` → when a payment fails.
- `subscription.create` → when a user subscribes to a plan.
- `subscription.disable` → when a subscription is cancelled.

These events are received automatically by your server and logged in your console.

Later, they can be saved in a database if needed.

HOW TO USE THE SYSTEM

All the demo pages are found in the `public/` folder.

You can open them in your browser after starting the app (default: <http://localhost:5001>).

Here's what each page does:

File Name	Description
`index.html`	The home page with links to all features
`create-plan.html`	Create a new plan on Paystack
`list-plans.html`	View all created plans
`create-subscription.html`	Subscribe a customer to a plan
`delete-subscription.html`	Cancel or disable a customer's subscription
`list-subscriptions.html`	View all active subscriptions
`list-transactions.html`	View all transactions (with pagination)
`test-payment.html`	Make a test payment
`verify-payment.html`	Verify a payment reference
`installment-form.html`	Create an installment payment plan
`view-installments.html`	View all active installments
`installment-dashboard.html`	Manage installment payments
`success.html`	Shown when payment succeeds
`failed.html`	Shown when payment fails
`webhook-test.html`	Used to test webhook events manually

Example usage:

1. Open `create-plan.html` → Fill in plan name, amount, and interval → Click "Create Plan"
2. Copy the generated Plan Code
3. Open `create-subscription.html` → Enter customer email and Plan Code → Click "Subscribe"
4. To cancel, go to `delete-subscription.html` and enter the same Plan Code

For installments:

1. Open `installment-form.html` → Enter customer email, total amount, and monthly payment → Click "Create Installment"
2. Open `view-installments.html` → Enter customer email → View all their active installments
3. The system will automatically charge customers when their payment is due

HOW WEBHOOKS WORK (IN SIMPLE TERMS)

Imagine Paystack as a messenger.

Whenever something happens (like a customer paying for a plan), Paystack sends a "message" to your app to notify it.

Your app listens to these messages through the `/api/webhooks` route.

To make sure the message is really from Paystack and not fake, your app checks a secret "signature" before trusting the data.

If the signature matches, it logs the event in the console like this:

 Webhook received: charge.success

If it doesn't match, it prints:

 Invalid Paystack signature

This helps keep your payment system safe and accurate.

EXAMPLE FLOW (REALISTIC SCENARIO)

Scenario 1: Creating a Subscription

1. You create a plan: "Gold Plan – ₦2000 monthly"
2. A customer subscribes to that plan.
3. Paystack charges them and sends a `charge.success` webhook.
4. Your app logs: "Charge successful".
5. If the customer cancels, Paystack sends a `subscription.disable` webhook.
6. Your app logs: "Subscription disabled".

7. You can open `list-subscriptions.html` to confirm the change.

Scenario 2: Installment Payment

1. A customer wants to buy something worth ₦50,000 but can't pay all at once.
2. You create an installment plan: ₦5,000 per month for 10 months.
3. The customer makes their first payment of ₦5,000.
4. Every month, the system automatically charges them ₦5,000.
5. After 10 months, the installment is fully paid and marked as "completed".
6. You can view all payment history in `view-installments.html`.

Scenario 3: Payment Verification

1. A customer makes a payment through your system.
2. They receive a payment reference (like a receipt number).
3. You can use `verify-payment.html` to check if the payment was successful.
4. Enter the reference number and click "Verify".
5. The system shows you all the payment details.

📧 EMAIL NOTIFICATIONS

The system can send email reminders to customers.

For example:

- When a subscription is about to expire (10 minutes before), the system sends an email reminder.
- The email tells the customer to make sure their payment method is active.

This helps prevent subscription interruptions.

🕒 AUTOMATED REMINDERS

The system runs automatic checks in the background:

1. Subscription Expiry Checker - Checks every minute for subscriptions that are about to expire and sends email reminders.

2. Installment Monitor - Checks every 5 minutes for installments that are due for payment and automatically charges customers.

You don't need to do anything manually; the system handles it automatically!

KEEPING THINGS SECURE

The system uses several security measures:

1. Webhook Signature Validation - Makes sure webhook messages are really from Paystack.
2. Environment Variables - Keeps sensitive information (like API keys) safe and hidden.
3. Database Storage - Saves all important information securely in a database.

TIPS FOR USING THE SYSTEM

1. **Always test with test keys first** - Use Paystack test keys when developing to avoid real charges.
2. **Check webhook logs** - Look at your console to see what webhook events are being received.
3. **Verify payments** - Always verify payments using the reference number before confirming orders.
4. **Monitor subscriptions** - Regularly check `list-subscriptions.html` to see which subscriptions are active.
5. **Track installments** - Use `view-installments.html` to see which customers have pending payments.

COMMON QUESTIONS

Q: What happens if a payment fails?

A: Paystack sends a `invoice.payment_failed` webhook, and your system logs it. You can then contact the customer or retry the payment.

Q: Can I change a plan after creating it?

A: Yes, you can update plans, but be careful as it might affect existing subscriptions.

Q: How do I know if a webhook was received?

A: Check your server console logs. You'll see messages like " Paystack Webhook Event: charge.success".

Q: What if a customer wants to pay their installment early?

A: They can make a manual payment through the system, and it will update their installment record automatically.

Q: How long does it take for webhooks to arrive?

A: Usually within a few seconds after the event happens.

 SUMMARY

This system helps you:

-  Create and manage payment plans
-  Handle customer subscriptions
-  Process one-time payments
-  Manage installment payments
-  Receive automatic notifications from Paystack
-  View all transactions and payments
-  Send email reminders to customers
-  Automatically charge customers when payments are due

Everything is designed to be simple and easy to use, even if you're not a technical expert!

