

Flutterwave Payment Gateway Integration for SmartPanel v4.1

Installation Guide

Prerequisites

- SmartPanel v4.1 (CodeIgniter framework)
 - cPanel/WHM hosting environment
 - MySQL/MariaDB database access
 - Flutterwave Test/Live API credentials
-

Package Contents

```
flutterwave_integration_package/  
├── flutterwave.php           # Main Flutterwave controller  
├── flutterwave_db_config.sql # Database configuration  
└── INSTALLATION_GUIDE.md    # This installation guide
```

Step 1: Upload Controller File

Via cPanel File Manager:

1. **Login to cPanel** → File Manager
2. **Navigate to:** `public_html/app/modules/add_funds/controllers/`

3. **Upload:** flutterwave.php

4. **Set Permissions:** 644 (rw-r--r--)

Via FTP/SFTP:

```
# Upload to: /app/modules/add_funds/controllers/flutterwave.php  
chmod 644 flutterwave.php
```

Step 2: Database Configuration

Method 1: Via phpMyAdmin (Recommended)

1. **Login to cPanel** → phpMyAdmin
2. **Select your SmartPanel database**
3. **Click "SQL" tab**
4. **Copy and paste** the contents of flutterwave_db_config.sql
5. **Update API Keys** in the SQL before execution:

```
sql -- Replace these with your actual Flutterwave credentials:  
"public_key":"FLWPUBK_TEST-your-actual-public-key"  
"secret_key":"FLWSECK_TEST-your-actual-secret-key"  
"encryption_key":"FLWSECK_TEST-your-actual-encryption-key"
```

6. **Click "Go"** to execute

Method 2: Via MySQL Command Line

```
mysql -u username -p database_name < flutterwave_db_config.sql
```

Step 3: Configure API Keys

Get Your Flutterwave Credentials:

1. **Login** to [Flutterwave Dashboard](#)
2. **Go to:** Settings → API Keys
3. **Copy:**
 - **Public Key** (FLWPUBK_TEST-xxx for test mode)
 - **Secret Key** (FLWSECK_TEST-xxx for test mode)
 - **Encryption Key** (optional, for webhook security)

Update Database Configuration:

Update the `payments` table with your actual API keys:

```
UPDATE payments
SET params = JSON_SET(
    params,
    '$.option.public_key', 'YOUR_ACTUAL_PUBLIC_KEY',
    '$.option.secret_key', 'YOUR_ACTUAL_SECRET_KEY',
    '$.option.encryption_key', 'YOUR_ACTUAL_ENCRYPTION_KEY'
)
WHERE type = 'flutterwave';
```

Step 4: Configuration Options

Available Settings (via Admin Panel or Database):

Setting	Default	Description
<code>environment</code>	<code>test</code>	<code>test</code> for sandbox, <code>live</code> for production
<code>tnx_fee</code>	<code>2.5</code>	Transaction fee percentage
<code>rate_to_usd</code>	<code>1500</code>	NGN to USD conversion rate
<code>min</code>	<code>100</code>	Minimum payment amount
<code>max</code>	<code>1000000</code>	Maximum payment amount

Update via SQL:

```
UPDATE payments
SET params = JSON_SET(
  params,
  '$.option.environment', 'live', -- Change to 'live' for
production
  '$.option.tnx_fee', '2.0',      -- Adjust fee percentage
  '$.option.rate_to_usd', '1600' -- Update exchange rate
)
WHERE type = 'flutterwave';
```

Step 5: Testing

Test Payment Flow:

1. **Login** to SmartPanel user account

2. **Go to:** Add Funds section
3. **Select:** "Flutterwave Payment Gateway"
4. **Enter:** Test amount (e.g., 1000 NGN)
5. **Click:** "Add Funds"
6. **Use Test Cards:**

```

```
Successful Payment
Card: 5531886652142950
CVV: 564
Expiry: 09/32
PIN: 3310
```

```
Failed Payment
Card: 5531886652142951
CVV: 564
Expiry: 09/32
PIN: 3310
```
```

Webhook Testing:

1. **Set Webhook URL** in Flutterwave Dashboard:

```
https://yourdomain.com/add_funds/flutterwave/webhook
```

2. **Test** webhook delivery via Flutterwave dashboard
-

Step 6: Going Live

Switch to Production:

1. **Get Live API Keys** from Flutterwave Dashboard

2. Update Database:

```
sql UPDATE payments SET params = JSON_SET( params,  
 '$.option.environment', 'live', '$.option.public_key',  
 'FLWPUBK_LIVE-your-live-public-key', '$.option.secret_key',  
 'FLWSECK_LIVE-your-live-secret-key' ) WHERE type = 'flutterwave';
```

3. Test with small live transaction

Optional Enhancements

Custom Logo:

Update the logo URL in the controller:

```
// In flutterwave.php, line ~87  
'logo' => base_url('assets/images/your-custom-logo.png')
```

Additional Currencies:

Add more currencies to the configuration:

```
UPDATE payments  
SET params = JSON_SET(  
    params,  
    '$.option.supported_currencies',  
    JSON_ARRAY('NGN', 'USD', 'EUR', 'GBP', 'KES', 'UGX', 'TZS',  
    'ZAR', 'GHS')  
)  
WHERE type = 'flutterwave';
```

Troubleshooting

Common Issues:

"Payment gateway not active"

Solution: Check `status` field in `payments` table:

```
UPDATE payments SET status = 1 WHERE type = 'flutterwave';
```

"API returned error"

Solutions:

1. Verify API keys are correct
2. Check internet connectivity
3. Ensure Flutterwave account is active
4. Check error logs: `app/logs/log-YYYY-MM-DD.log`

Webhook not working

Solutions:

1. Verify webhook URL in Flutterwave dashboard
2. Check server logs for incoming requests
3. Ensure webhook signature verification is correct

Currency conversion issues

Solution: Update exchange rate:

```
UPDATE payments
SET params = JSON_SET(params, '$.option.rate_to_usd', '1600')
WHERE type = 'flutterwave';
```

Debug Mode:

Enable CodeIgniter logging in `app/config/config.php`:

```
$config['log_threshold'] = 2; // Enable error and debug logs
```

Support

Flutterwave Resources:

- **Documentation:** <https://developer.flutterwave.com/docs>
- **Support:** <https://support.flutterwave.com>
- **Dashboard:** <https://dashboard.flutterwave.com>

SmartPanel Integration:

- Check `app/logs/` for error logs
 - Verify database configuration
 - Test with Flutterwave's test cards
-

Verification Checklist

Before going live, ensure:

- Controller file uploaded to correct location
- Database configuration imported successfully
- API keys updated with actual credentials
- Test payment completed successfully
- Webhook URL configured in Flutterwave dashboard
- Transaction logs created in database

- User balance updated after successful payment
 - Email notifications working (if configured)
 - Error handling tested with failed payments
 - Production environment variables set
-

Security Notes

1. **API Keys:** Keep secret keys secure, never expose in frontend
 2. **Webhook Verification:** Enable signature verification for production
 3. **SSL Certificate:** Ensure HTTPS is enabled for payment pages
 4. **Amount Verification:** Always verify payment amounts server-side
 5. **Regular Updates:** Keep API keys rotated as per Flutterwave recommendations
-

Installation Complete!

Your SmartPanel now supports Flutterwave payments with multi-currency and multi-payment method support.

Generated by MiniMax Agent for SmartPanel v4.1 - `${new Date().toISOString().split('T')[0]}`