X (Twitter) & Pinterest Integration Guide

Overview

Your Social Media Automation API now supports posting to **X (Twitter)** and **Pinterest** in addition to Facebook and Instagram.

X (Twitter) Integration

Platform Features

- ✅ Tweet posting with media (up to 4 images)
- ✅ Thread creation (multiple connected tweets)
- ✅ Alt text for accessibility
- ✅ 280 character limit enforcement
- ✅ OAuth 2.0 with PKCE (secure authentication)
- ✅ Automatic token refresh

Image Requirements

Post Type	Dimensions	Aspect Ratio	Max Size
Tweet	1200x675	16:9	5MB
Square	1200x1200	1:1	5MB
Header	1500x500	3:1	5MB

Supported Formats: JPEG, PNG, GIF
Max Images per Tweet: 4

OAuth Setup

1. Create X Developer Account

1. Go to developer.twitter.com
2. Apply for developer access
3. Create a new app in the Developer Portal

2. Configure App Settings

```
App Type: Web App
Callback URL: https://yourdomain.com/api/v1/auth/x/callback
Website URL: https://yourdomain.com
Terms of Service: https://yourdomain.com/terms
Privacy Policy: https://yourdomain.com/privacy
```

3. Required Permissions

- ✅ Read and Write (tweet.read, tweet.write)
- ✅ Users (users.read)
- ✅ Offline Access (offline.access) - for refresh tokens

4. Environment Variables

bash

```bash
X_CLIENT_ID=your_client_id
X_CLIENT_SECRET=your_client_secret
X_REDIRECT_URI=https://yourdomain.com/api/v1/auth/x/callback
```

API Usage

Connect X Account

python

```python
# User initiates OAuth
GET /api/v1/auth/connect/x?user_id=user_123

# User authorizes on X
# Callback handled automatically

# Account is now connected!
```

Post Tweet with Image

python

```python
import requests

# 1. Upload image
with open('image.jpg', 'rb') as f:
    upload_response = requests.post(
        'https://api.yourdomain.com/api/v1/upload',
        files={'file': f},
        headers={'Authorization': 'Bearer YOUR_API_KEY'}
    )
image_id = upload_response.json()['image_id']

# 2. Create post
post_data = {
    'platforms': ['x'],
    'post_types': {'x': 'tweet'},
    'caption': 'Check out this amazing photo! 📷 #photography',
    'image_ids': [image_id],
    'auto_hashtags': True
}

response = requests.post(
    'https://api.yourdomain.com/api/v1/posts',
    json=post_data,
    headers={'Authorization': 'Bearer YOUR_API_KEY'}
)

result = response.json()
print(f"Tweet posted! URL: {result['platform_results']['x']['url']}")
```

Post Tweet Thread

📋 ✓

python

```
# For longer content, automatically split into thread
post_data = {
    'platforms': ['x'],
    'post_types': {'x': 'thread'},
    'caption': '''This is a long story that will be split into multiple tweets.

    Each part will be connected as a thread.

    The API handles the complexity of posting threads automatically.''',
    'image_ids': [image_id],
    'thread_auto_split': True  # Automatically split at 280 chars
}
```

Character Limit Handling

The API automatically handles X's 280 character limit:

python

```
# Option 1: Truncate with ellipsis
post_data = {
    'caption': 'Very long text...',
    'x_options': {
        'truncate': True,
        'truncate_suffix': '... (more)'
    }
}

# Option 2: Auto-thread (split into multiple tweets)
post_data = {
    'caption': 'Very long text...',
    'x_options': {
        'auto_thread': True,
        'thread_numbering': True  # Adds 1/3, 2/3, etc.
    }
}
```

Rate Limits (X Platform)

User Limits:

- 300 tweets per 3 hours
- 50 tweets per day (for new accounts)

API Rate Limits:

- 200 requests per 15 minutes (per user)
- 450 requests per 15 minutes (per app)

Pinterest Integration

Platform Features

- ✅ Pin creation with images
- ✅ Board management
- ✅ Idea pins (multi-page pins)
- ✅ Shopping pins (product catalogs)
- ✅ Video pins support
- ✅ Rich pins with metadata

Image Requirements

Post Type	Dimensions	Aspect Ratio	Max Size
Pin	1000x1500	2:3	20MB
Square Pin	1000x1000	1:1	20MB
Wide Pin	1000x500	2:1	20MB

Optimal Dimensions: 1000x1500 (2:3 ratio)
Minimum Width: 600px
Supported Formats: JPEG, PNG
Max Images per Pin: 1 (or 5 for Idea pins)

OAuth Setup

1. Create Pinterest App

1. Go to developers.pinterest.com
2. Create a new app
3. Complete app verification

2. Configure App Settings

```
App Name: Your App Name
Description: Social media automation tool
Redirect URI: https://yourdomain.com/api/v1/auth/pinterest/callback
Website: https://yourdomain.com
```

3. Required Scopes

- ✅ boards:read
- ✅ boards:write
- ✅ pins:read
- ✅ pins:write
- ✅ user_accounts:read

4. Environment Variables

bash

```bash
PINTEREST_APP_ID=your_app_id
PINTEREST_APP_SECRET=your_app_secret
PINTEREST_REDIRECT_URI=https://yourdomain.com/api/v1/auth/pinterest/callback
```

API Usage

Connect Pinterest Account

python

```
# User initiates OAuth
GET /api/v1/auth/connect/pinterest?user_id=user_123

# User authorizes on Pinterest
# Callback returns connected boards

# Response includes:
{
  "success": true,
  "username": "myusername",
  "boards": [
    {"id": "board_123", "name": "Travel", "pin_count": 45},
    {"id": "board_456", "name": "Food", "pin_count": 28}
  ]
}
```

Create Pin

📋 ✓

python

```python
import requests

# 1. Upload image
with open('image.jpg', 'rb') as f:
    upload_response = requests.post(
        'https://api.yourdomain.com/api/v1/upload',
        files={'file': f},
        headers={'Authorization': 'Bearer YOUR_API_KEY'}
    )
image_id = upload_response.json()['image_id']

# 2. Create pin
post_data = {
    'platforms': ['pinterest'],
    'post_types': {'pinterest': 'pin'},
    'pinterest_board_id': 'board_123',  # Required for Pinterest
    'caption': 'Amazing sunset photo! Perfect for your inspiration board.',
    'title': 'Beautiful Sunset',  # Pinterest requires title
    'link': 'https://yourwebsite.com/blog/sunset-photography',  # Optional
    'image_ids': [image_id],
    'auto_hashtags': True
}

response = requests.post(
    'https://api.yourdomain.com/api/v1/posts',
    json=post_data,
    headers={'Authorization': 'Bearer YOUR_API_KEY'}
)

result = response.json()
print(f"Pin created! URL: {result['platform_results']['pinterest']['url']}")
```

List User Boards

python

```
# Get user's boards to let them choose where to pin
response = requests.get(
    'https://api.yourdomain.com/api/v1/pinterest/boards',
    headers={'Authorization': 'Bearer YOUR_API_KEY'}
)

boards = response.json()['boards']
for board in boards:
    print(f"{board['name']} - {board['pin_count']} pins")
```

Create New Board

python

```
board_data = {
    'name': 'My New Board',
    'description': 'A collection of beautiful photos',
    'privacy': 'PUBLIC'  # or 'SECRET'
}

response = requests.post(
    'https://api.yourdomain.com/api/v1/pinterest/boards',
    json=board_data,
    headers={'Authorization': 'Bearer YOUR_API_KEY'}
)

board_id = response.json()['board_id']
```

Pinterest Best Practices

Title Guidelines

- Keep under 100 characters
- Include keywords
- Be descriptive and specific
- Use title case

Good: "Easy Vegan Dinner Recipes for Busy Weeknights"
Bad: "dinner recipes"

Description Guidelines

- Max 500 characters
- Include relevant keywords
- Add call-to-action
- Use 3-5 hashtags

Example:

```
Delicious and easy vegan dinner recipes perfect for busy weeknights!
Quick prep, healthy ingredients, family-friendly.
Get the full recipe on our website! 🌱

#VeganRecipes #HealthyDinner #PlantBased
```

Link Strategy

- Always include a destination link
- Link to relevant blog posts or products
- Use UTM parameters for tracking
- Ensure links are working

Rate Limits (Pinterest Platform)

User Limits:

- 200 pins per day (recommended)
- No strict API rate limits for posting

API Rate Limits:

- 200 requests per hour per access token
- 10,000 requests per day per app

Multi-Platform Posting

Post to All Platforms Simultaneously

python

```
# Upload image once
upload_response = requests.post(
    'https://api.yourdomain.com/api/v1/upload',
    files={'file': open('image.jpg', 'rb')},
    headers={'Authorization': 'Bearer YOUR_API_KEY'}
)
image_id = upload_response.json()['image_id']

# Post to all 4 platforms
post_data = {
    'platforms': ['facebook', 'instagram', 'x', 'pinterest'],
    'post_types': {
        'facebook': 'feed',
        'instagram': 'feed_square',
        'x': 'tweet',
        'pinterest': 'pin'
    },
    'caption': 'Amazing photo! Check it out! #photography',
    'title': 'Amazing Photo',  # For Pinterest
    'pinterest_board_id': 'board_123',  # Required for Pinterest
    'image_ids': [image_id],
    'auto_hashtags': True,
    'scheduled_for': '2025-01-20T18:00:00Z'  # Optional scheduling
}

response = requests.post(
    'https://api.yourdomain.com/api/v1/posts',
    json=post_data,
    headers={'Authorization': 'Bearer YOUR_API_KEY'}
)

# Check results for each platform
result = response.json()
for platform, details in result['platform_results'].items():
    if details['status'] == 'success':
        print(f"✅ {platform}: {details['url']}")
    else:
        print(f"❌ {platform}: {details['error']}")
```

Platform-Specific Customization

python

```python
# Customize caption for each platform
post_data = {
    'platforms': ['x', 'pinterest'],
    'post_types': {
        'x': 'tweet',
        'pinterest': 'pin'
    },
    'captions': {
        'x': 'Short tweet-friendly caption! 📷 #photo',
        'pinterest': 'Longer, more descriptive Pinterest caption with keywords and details about the image.'
    },
    'titles': {
        'pinterest': 'Professional Photography Tips'
    },
    'pinterest_board_id': 'board_123',
    'image_ids': [image_id]
}
```

Updated Environment Variables

Add these to your `.env` file:

bash

```bash
# X (Twitter) Configuration
X_CLIENT_ID=your_x_client_id
X_CLIENT_SECRET=your_x_client_secret
X_REDIRECT_URI=https://yourdomain.com/api/v1/auth/x/callback
X_CONSUMER_KEY=your_consumer_key  # OAuth 1.0a (for posting)
X_CONSUMER_SECRET=your_consumer_secret

# Pinterest Configuration
```