

# Instagram DM Integration Guide

---

## Overview

This guide covers the complete Instagram Direct Messaging (DM) integration in the Soshogle CRM system. The integration allows you to:

- Receive Instagram direct messages in real-time
- Send messages to Instagram users
- Manage Instagram conversations within the CRM
- Automatically create leads from Instagram DMs
- Connect multiple Instagram accounts

## Architecture

### Key Components

1. **Webhook Endpoint:** `/api/instagram/webhook`
  - Receives incoming messages from Instagram
  - Handles webhook verification from Meta
  - Processes message events and stores them in the database
2. **OAuth Flow:** `/api/instagram/oauth`
  - Initiates Instagram OAuth authentication
  - Handles callback and token exchange
  - Stores access tokens securely
3. **Instagram Service:** `lib/messaging-sync/instagram-service.ts`
  - Handles message sending
  - Processes incoming webhook data
  - Fetches user information from Instagram API
4. **Database Models:**
  - `ChannelConnection` : Stores Instagram account connections
  - `Conversation` : Manages Instagram DM conversations
  - `ConversationMessage` : Stores individual messages
  - `Lead` : Automatically created from new conversations

## Setup Instructions

### 1. Create Instagram App in Meta Developer Console

1. Go to <https://developers.facebook.com/apps/> (<https://developers.facebook.com/apps/>)
2. Click “Create App”
3. Select “Business” as the app type
4. Fill in your app details
5. Add “Instagram” product to your app

## 2. Configure App Credentials

Add the following environment variables to your `.env` file:

```
INSTAGRAM_APP_ID=your_app_id
INSTAGRAM_APP_SECRET=your_app_secret
INSTAGRAM_VERIFY_TOKEN=your_custom_verify_token
NEXT_PUBLIC_INSTAGRAM_VERIFY_TOKEN=your_custom_verify_token
```

**Generate a secure verify token:**

```
node -e "console.log(require('crypto').randomBytes(32).toString('hex'))"
```

## 3. Configure Webhook in Meta Developer Console

1. Navigate to your Instagram app in the Meta Developer Console
2. Go to **Products → Instagram → Webhooks**
3. Click **Edit Subscription** (or **Add Webhook** if first time)
4. Enter the following:

**Callback URL:**

`https://nexrel.soshogleagents.com/api/instagram/webhook`

**Verify Token:**

[Your `INSTAGRAM_VERIFY_TOKEN` from `.env`]

1. Subscribe to the following webhook fields:
  - `messages` (required for DMs)
  - `messaging_postbacks` (for button interactions)
  - `messaging_optins` (for opt-in events)
2. Click **Verify and Save**

## 4. App Review (Required for Production)

For production use, you need to submit your app for App Review:

1. Request the following permissions:
  - `instagram_basic`
  - `instagram_manage_messages`
  - `pages_show_list`
  - `pages_messaging`
2. Provide use case details explaining how you'll use Instagram messaging
3. Submit screencast video demonstrating the feature
4. Wait for approval (usually 3-5 business days)

## Testing

### Using the Built-in Test Tool

1. Log in to your CRM
2. Navigate to **Dashboard → Soshogle Multi-Channel**

3. Click the **Webhook Test** tab

4. Click **Run Webhook Tests**

The tool will verify:

- Webhook verification endpoint is responding correctly
- Message handling endpoint is processing events

## Manual Testing

### Test Webhook Verification

```
curl "https://nexrel.soshogleagents.com/api/instagram/webhook?
hub.mode=subscribe&hub.verify_token=YOUR_VERIFY_TOKEN&hub.challenge=test_123"
```

Expected response: `test_123` (echoes back the challenge)

### Test Message Handling

```
curl -X POST https://nexrel.soshogleagents.com/api/instagram/webhook \
-H "Content-Type: application/json" \
-d '{
  "object": "instagram",
  "entry": [
    {
      "id": "123456789",
      "time": 1234567890,
      "messaging": [
        {
          "sender": { "id": "test_user" },
          "recipient": { "id": "your_page_id" },
          "timestamp": 1234567890,
          "message": {
            "mid": "msg_123",
            "text": "Hello"
          }
        }
      ]
    }
  ]
}'
```

Expected response: `{"received":true}`

## Usage

### Connecting an Instagram Account

1. Navigate to **Soshogle Multi-Channel** page
2. Click **Connect Instagram**
3. Log in with your Instagram account
4. Authorize the app to manage messages
5. Select the Instagram Business account to connect

### Viewing Instagram Conversations

1. Go to **Dashboard → Conversations**
2. Filter by **Instagram** channel
3. Click on any conversation to view and respond to messages

## Sending Messages

Messages can be sent through:

1. **Conversation View:** Reply directly in the conversation thread
2. **API Endpoint:** Use `/api/instagram/messages/send`

```
const response = await fetch('/api/instagram/messages/send', {
  method: 'POST',
  headers: { 'Content-Type': 'application/json' },
  body: JSON.stringify({
    recipientId: 'instagram_user_id',
    message: 'Hello from Soshogle!',
  }),
});
```

## API Endpoints

### Webhook Endpoints

`GET /api/instagram/webhook`

Handles webhook verification from Meta.

#### Query Parameters:

- `hub.mode` : Must be “subscribe”
- `hub.verify_token` : Your verify token
- `hub.challenge` : Random string to echo back

#### Response:

- Returns the challenge string if verification succeeds
- Returns 403 if verification fails

`POST /api/instagram/webhook`

Receives Instagram messaging events.

#### Request Body:

```
{
  "object": "instagram",
  "entry": [
    {
      "id": "instagram_account_id",
      "time": 1234567890,
      "messaging": [
        {
          "sender": { "id": "sender_id" },
          "recipient": { "id": "recipient_id" },
          "timestamp": 1234567890,
          "message": {
            "mid": "message_id",
            "text": "message content"
          }
        }
      ]
    }
  ]
}
```

**Response:**

```
{ "received": true }
```

**OAuth Endpoints****GET /api/instagram/oauth**

Initiates Instagram OAuth flow.

**Response:**

```
{
  "authUrl": "https://instagram.com/oauth/authorize?..."
}
```

**GET /api/instagram/oauth/callback**

Handles OAuth callback.

**Query Parameters:**

- code : Authorization code from Instagram

**Response:**

Redirects to dashboard with success/error status

**Messaging Endpoints****POST /api/instagram/messages/send**

Sends a message to an Instagram user.

**Request Body:**

```
{
  "recipientId": "instagram_user_id",
  "message": "Hello!",
  "attachmentUrl": "https://i.ytimg.com/vi/sJTrvoeKUmk/hq720.jpg?sqp=-oaymwEhCK4-
FEIIDSFryq4qpAxMIARUAAAAGAElaADIQj0AgKJD&rs=A0n4CLaqiDhjHmfac0CQkVHduXIqDe0jdg" //
optional
}
```

**Response:**

```
{
  "success": true,
  "messageId": "msg_123"
}
```

**GET /api/instagram/status**

Checks Instagram connection status.

**Response:**

```
{
  "connected": true,
  "accountName": "@your_instagram",
  "accountId": "123456789"
}
```

## Webhook Event Types

### Messages

#### Text Message:

```
{
  "sender": { "id": "user_id" },
  "recipient": { "id": "page_id" },
  "timestamp": 1234567890,
  "message": {
    "mid": "msg_id",
    "text": "Hello"
  }
}
```

#### Image Message:

```
{
  "sender": { "id": "user_id" },
  "recipient": { "id": "page_id" },
  "timestamp": 1234567890,
  "message": {
    "mid": "msg_id",
    "attachments": [
      {
        "type": "image",
        "payload": {
          "url": "https://..."
        }
      }
    ]
  }
}
```

## Troubleshooting

### Webhook Not Receiving Messages

#### 1. Check webhook configuration in Meta Developer Console:

- Verify URL is correct and uses HTTPS
- Verify token matches your .env file
- Ensure subscriptions are active

#### 2. Test webhook endpoint:

```
bash
```

```
curl "https://nexrel.soshogleagents.com/api/instagram/webhook?
hub.mode=subscribe&hub.verify_token=YOUR_TOKEN&hub.challenge=test"
```

### 3. Check server logs:

- Look for webhook events in application logs
- Verify no errors during message processing

## OAuth Connection Fails

### 1. Verify app credentials:

- Check `INSTAGRAM_APP_ID` and `INSTAGRAM_APP_SECRET` in .env
- Ensure credentials match those in Meta Developer Console

### 2. Check redirect URIs:

- In Meta Developer Console, add your callback URL to valid OAuth redirect URIs:  
`https://nexrel.soshogleagents.com/api/instagram/oauth/callback`

### 3. Verify app is in live mode:

- Test accounts can only be used in development mode
- Production requires app review approval

## Messages Not Appearing in CRM

### 1. Check database connection:

- Verify `DATABASE_URL` in .env is correct
- Check database logs for errors

### 2. Verify channel connection:

- Go to Soshogle Multi-Channel page
- Ensure Instagram account shows as "Connected"
- Try disconnecting and reconnecting

### 3. Check conversation creation:

- Messages should automatically create conversations
- Verify conversation appears in database

## Rate Limiting

Instagram API has rate limits:

- **Messages:** 250 per hour per user
- **API calls:** Varies by endpoint

### Best Practices:

- Implement exponential backoff for retries
- Cache user information to reduce API calls
- Monitor rate limit headers in API responses

## Security Considerations

---

### 1. Webhook Verification:

- Always verify `hub.verify_token` before responding
- Use a strong, randomly generated verify token
- Never commit tokens to version control

### 2. Access Token Storage:

- Tokens are encrypted in the database
- Tokens are never exposed to client-side code
- Implement token refresh mechanism

### 3. HTTPS Required:

- All webhook URLs must use HTTPS
- Meta will reject HTTP endpoints

### 4. Data Privacy:

- Follow Instagram's Platform Terms
- Only store necessary user data
- Implement data retention policies
- Provide data deletion mechanisms

## Database Schema

### ChannelConnection

```
model ChannelConnection {}  
  id          String  @id @default(cuid())  
  userId      String  
  channelType String  // "INSTAGRAM"  
  providerType String  // "INSTAGRAM"  
  channelIdentifier String? // Instagram account ID  
  displayName  String? // @username  
  accessToken  String? // Encrypted access token  
  refreshToken String? // For token refresh  
  providerAccountId String? // Instagram business account ID  
  status       String  // "CONNECTED", "DISCONNECTED"  
  isActive     Boolean @default(true)  
  metadata     Json?   // Additional account info  
  lastSyncedAt DateTime?  
  createdAt    DateTime @default(now())  
  updatedAt    DateTime @updatedAt  
}
```

### Conversation

```
model Conversation {}  
  id          String  @id @default(cuid())  
  userId      String  
  channelConnectionId String  
  contactIdentifier String  // Instagram user ID  
  contactName   String  // Display name/username  
  contactAvatar  String? // Profile picture URL  
  status        String  // "ACTIVE", "ARCHIVED"  
  lastMessageAt DateTime  
  lastMessagePreview String?  
  unreadCount   Int    @default(0)  
  createdAt    DateTime @default(now())  
  updatedAt    DateTime @updatedAt  
}
```

## ConversationMessage

```
model ConversationMessage {
    id          String      @id @default(cuid())
    conversationId String
    userId       String
    direction    String     // "INBOUND", "OUTBOUND"
    content      String
    attachments  Json?      // Array of attachment objects
    status       String     // "SENT", "DELIVERED", "READ", "FAILED"
    externalMessageId String? // Instagram message ID
    providerData  Json?      // Original webhook data
    createdAt    DateTime   @default(now())
}
```

## Best Practices

### 1. Response Time:

- Respond to webhooks within 20 seconds
- Process heavy operations asynchronously
- Return 200 OK immediately

### 2. Error Handling:

- Log all errors with context
- Implement retry logic for failed operations
- Monitor webhook failures

### 3. Testing:

- Use test accounts during development
- Test with various message types (text, images, etc.)
- Verify lead creation and conversation management

### 4. Monitoring:

- Track webhook delivery success rate
- Monitor message processing times
- Set up alerts for failures

## Resources

- [Instagram Messaging API Documentation](https://developers.facebook.com/docs/messenger-platform/instagram) (<https://developers.facebook.com/docs/messenger-platform/instagram>)
- [Instagram Webhooks Documentation](https://developers.facebook.com/docs/instagram/webhooks/) (<https://developers.facebook.com/docs/instagram/webhooks/>)
- [Instagram Graph API Reference](https://developers.facebook.com/docs/instagram-api) (<https://developers.facebook.com/docs/instagram-api>)
- [Meta App Review Guidelines](https://developers.facebook.com/docs/app-review) (<https://developers.facebook.com/docs/app-review>)

## Support

For issues or questions:

1. Check this documentation
2. Review Meta Developer Console logs
3. Test using the built-in webhook test tool
4. Contact the development team