

CX-Catalyst - Email Integration Guide

Complete guide for configuring email (Gmail) as a support intake channel and notification delivery method.

Table of Contents

1. Overview
 2. Prerequisites
 3. Gmail OAuth Setup
 4. n8n Credential Configuration
 5. Inbound Email Processing
 6. Outbound Email Delivery
 7. Email Templates
 8. Workflow Integration
 9. Troubleshooting
-

Overview

CX-Catalyst uses Gmail integration for two purposes:

- **Inbound:** Receive support requests via email and route them into the AI triage pipeline
- **Outbound:** Send solution emails to customers, leadership reports, and escalation notifications

Email is an **optional** integration – the core system uses webhooks and Slack as primary channels. Adding email provides an additional customer-facing intake channel and a delivery mechanism for reports.

Prerequisites

- ⓘ Google Workspace or Gmail account dedicated to support (e.g., support@company.com)
 - ⓘ Google Cloud Console access to create OAuth credentials
 - ⓘ n8n instance accessible via HTTPS (required for OAuth callback)
-

Gmail OAuth Setup

Step 1: Create a Google Cloud Project

1. Go to Google Cloud Console
2. Click **Select a project > New Project**
3. Name: CX-Catalyst Support
4. Click **Create**

Step 2: Enable the Gmail API

1. In the Cloud Console, go to **APIs & Services > Library**
2. Search for **Gmail API**
3. Click **Enable**

Step 3: Configure OAuth Consent Screen

1. Go to **APIs & Services > OAuth consent screen**
2. Select **Internal** (for Google Workspace) or **External**
3. Fill in:
 - **App name:** CX-Catalyst Support System
 - **User support email:** Your admin email
 - **Authorized domains:** Your company domain
4. Add scopes:
 - `https://www.googleapis.com/auth/gmail.send`
 - `https://www.googleapis.com/auth/gmail.readonly`
 - `https://www.googleapis.com/auth/gmail.modify` (for marking emails as read)
5. Save

Step 4: Create OAuth Credentials

1. Go to **APIs & Services > Credentials**
 2. Click **Create Credentials > OAuth client ID**
 3. Application type: **Web application**
 4. Name: n8n CX-Catalyst
 5. Authorized redirect URI: `https://your-n8n-instance.com/rest/oauth2-credential/callback`
 6. Click **Create**
 7. Copy the **Client ID** and **Client Secret**
-

n8n Credential Configuration

Create Gmail Credential

1. In n8n, go to **Settings > Credentials**
2. Click **Add Credential > Gmail OAuth2 API**
3. Enter:
 - **Client ID:** From Google Cloud Console
 - **Client Secret:** From Google Cloud Console
4. Click **Connect my account**
5. Sign in with the support email account
6. Grant the requested permissions
7. Click **Save**

Verify Connection

After saving, test the credential by: 1. Creating a temporary workflow with a **Gmail** node 2. Selecting the credential 3. Setting action to **Get Many** messages 4. Executing – you should see recent emails

Inbound Email Processing

How It Works

Customer sends email to support@company.com



Gmail Trigger node polls for new emails



Extract: sender, subject, body, attachments



Transform to standard intake format



Pass to Workflow 1: Smart Intake & Triage



AI classifies, routes, and resolves

Gmail Trigger Configuration

In Workflow 1 (or a dedicated email intake workflow):

1. Add a **Gmail Trigger** node
2. Set **Poll Times**: Every 1 minute
3. Set **Mailbox Label**: Apply a label like support-inbox to filter relevant emails
4. Set **Read Status**: Unread only
5. Connect to a **Code** node that transforms the email into the intake format:

```
// Transform email to intake format
const email = $input.first().json;

return [
  json: {
    customer_email: email.from.value[0].address,
    customer_name: email.from.value[0].name || email.from.value[0].address,
    description: `Subject: ${email.subject}\n\n${email.textPlain} ||
${email.snippet}`,
    severity: "medium",
    channel: "email",
    metadata: {
      email_id: email.id,
      email_thread_id: email.threadId,
```

```

        email_subject: email.subject
    }
}
];

```

6. Connect the output to the existing intake triage pipeline

Email Parsing Considerations

- **Subject line** – Append to the description so the AI can use it for classification
 - **HTML body** – Use `textPlain` over `textHtml` for cleaner AI input
 - **Attachments** – Log attachment names in metadata; attachment content is not sent to AI
 - **Thread replies** – Check `threadId` to associate follow-up emails with existing cases
-

Outbound Email Delivery

Solution Delivery

When a case is resolved (self-service or human-approved), send the solution via email:

1. Add a **Gmail** node at the end of the resolution branch
2. Set **Operation**: Send
3. Set **To**: `{{ $json.customer_email }}`
4. Set **Subject**: Re: `{{ $json.email_subject || "Your Support Request" }}` [Case `{{ $json.case_id }}`]
5. Set **Message**: Use the HTML template (see Email Templates below)

Leadership Reports

Workflow 5 sends daily leadership reports via email:

- **To**: `{{ $env.LEADERSHIP_EMAIL }}`
- **Subject**: CX-Catalyst Daily Report - `{{ now.format('YYYY-MM-DD') }}`
- **Body**: HTML-formatted report with case metrics, trends, and AI performance

Escalation Notifications

For critical escalations without Slack:

- **To**: `{{ $env.SUPPORT_TEAM_EMAIL }}`
 - **Subject**: [URGENT] Case `{{ $json.case_id }}` - `{{ $json.priority }}` Priority Escalation
 - **Body**: Case details, AI analysis, and escalation reason
-

Email Templates

Solution Email Template

```

<div style="font-family: Arial, sans-serif; max-width: 600px;">
    <h2>Support Case Resolution</h2>
    <p>Hi {{ $json.customer_name }},</p>

```

```

<p>We've reviewed your support request and have a solution for you:</p>

<div style="background: #f5f5f5; padding: 16px; border-radius: 8px; margin: 16px 0;">
  <strong>Case ID:</strong> {{ $json.case_id }}<br>
  <strong>Category:</strong> {{ $json.category }}<br>
  <strong>Priority:</strong> {{ $json.priority }}
</div>

<h3>Solution</h3>
{{ $json.solution_html }}

<h3>Referenced Articles</h3>
<ul>
  {{ $json.kb_articles }}
</ul>

<p>If this resolves your issue, no further action is needed. If you need additional help, please reply to this email or submit a new request.</p>

<p>
  <a href="{{ $json.feedback_url }}?score=5">Rate this solution</a>
</p>

<p>Best regards,<br>Support Team</p>
</div>

```

Escalation Email Template

```

<div style="font-family: Arial, sans-serif; max-width: 600px;">
  <h2 style="color: #d32f2f;">Escalation Alert</h2>

  <div style="background: #fff3e0; padding: 16px; border-radius: 8px; border-left: 4px solid #ff9800;">
    <strong>Case:</strong> {{ $json.case_id }}<br>
    <strong>Customer:</strong> {{ $json.customer_name }}  

    {{ $json.account_tier }}<br>
    <strong>Priority:</strong> {{ $json.priority }}<br>
    <strong>Reason:</strong> {{ $json.escalation_reason }}
  </div>

  <h3>Issue Description</h3>
  <p>{{ $json.description }}</p>

  <h3>AI Analysis</h3>
  <p>{{ $json.ai_reasoning }}</p>

  <p><a href="{{ $json.review_url }}">Review this case</a></p>
</div>

```

Workflow Integration

Which Workflows Use Email

Workflow	Email Function	Direction
WF1: Intake & Triage	Receive support requests	Inbound
WF2: Self-Service	Send automated solutions	Outbound
WF4: Collaborative	Send human-approved solutions	Outbound
WF5: Learning	Send daily leadership reports	Outbound

Environment Variables

SUPPORT_EMAIL=support@company.com

LEADERSHIP_EMAIL=leadership@company.com

SUPPORT_TEAM_EMAIL=support-team@company.com

Troubleshooting

Issue	Cause	Solution
OAuth flow fails	Redirect URI mismatch	Verify the n8n callback URL matches exactly in Google Cloud Console
“Insufficient permissions”	Missing scopes	Re-authorize with gmail.send, gmail.readonly, gmail.modify scopes
Emails not detected	Wrong label filter	Check the Gmail Trigger label matches the inbox label
Duplicate processing	Emails re-read	Ensure the trigger marks processed emails as read
Send rate limits	Gmail daily send limit	Gmail allows 500 emails/day (Workspace: 2,000/day); use batch delays if needed
HTML not rendering	Template errors	Test templates with a manual send before deploying in workflows
“Token expired”	OAuth refresh failed	Re-connect the Gmail credential in n8n; check that the Google Cloud app is not suspended
