Gmail API Setup Guide

This guide will help you set up Gmail API for sending emails in your ROI Calculator application.

📋 Prerequisites

- A Google account
- Access to Google Cloud Console
- Your project already has `googleapis` and `nodemailer` installed

🚀 Step-by-Step Setup

1. Create Google Cloud Project

1. Go to Google Cloud Console
2. Click **"Select a project"** → **"New Project"**
3. Enter project name (e.g., "Robotipy Email Service")
4. Click **"Create"**

2. Enable Gmail API

1. In Google Cloud Console, go to **"APIs & Services"** → **"Library"**
2. Search for **"Gmail API"**
3. Click on **"Gmail API"** and press **"Enable"**

3. Create OAuth2 Credentials

1. Go to **"APIs & Services"** → **"Credentials"**
2. Click **"+ CREATE CREDENTIALS"** → **"OAuth 2.0 Client IDs"**
3. If prompted, configure OAuth consent screen:
   - Choose **"External"** user type
   - Fill in required fields:
     - App name: "Robotipy ROI Calculator"
     - User support email: your email
     - Developer contact email: your email
   - Add scopes: `https://www.googleapis.com/auth/gmail.send`
   - Add test users (your email)
4. Create OAuth 2.0 Client ID:
   - Application type: **"Web application"**
   - Name: "Robotipy Gmail Client"
   - Authorized redirect URIs: `https://developers.google.com/oauthplayground`
   - Click **"Create"**
5. **Save** the Client ID and Client Secret

4. Get Refresh Token

1. Go to OAuth 2.0 Playground

2. Click the **gear icon** (settings) in the top right
3. Check **"Use your own OAuth credentials"**
4. Enter your **Client ID** and **Client Secret**
5. In **"Step 1"**, find **"Gmail API v1"** → select `https://www.googleapis.com/auth/gmail.send`
6. Click **"Authorize APIs"**
7. Sign in with your Google account and grant permissions
8. In **"Step 2"**, click **"Exchange authorization code for tokens"**
9. **Copy the Refresh Token** (starts with `1//`)

5. Configure Environment Variables

Create or update your `.env.local` file with:

```
# Gmail API Configuration
GMAIL_CLIENT_ID=your-client-id.googleusercontent.com
GMAIL_CLIENT_SECRET=your-client-secret
GMAIL_REFRESH_TOKEN=your-refresh-token
GMAIL_USER_EMAIL=your-email@gmail.com
GMAIL_REDIRECT_URL=https://developers.google.com/oauthplayground
```

6. Test the Configuration

Create a test API route to verify your setup:

```
// app/api/test-email/route.js
import { sendGmailEmail, verifyGmailConnection } from "@/libs/gmail";
import { NextResponse } from "next/server";

export async function GET() {
  try {
    // Test connection
    const connectionTest = await verifyGmailConnection();

    if (!connectionTest.success) {
      return NextResponse.json({
        error: "Gmail API connection failed",
        details: connectionTest.message
      }, { status: 500 });
    }

    // Send test email
    const result = await sendGmailEmail({
      to: process.env.GMAIL_USER_EMAIL,
      subject: "Test Email from ROI Calculator",
      html: "<h1>Gmail API is working!</h1><p>Your ROI calculator can now send emails.</p>"
    });

    return NextResponse.json({
      success: true,
      message: "Test email sent successfully",
      messageId: result.messageId
    });

  } catch (error) {
    return NextResponse.json({
      error: "Failed to send test email",
      details: error.message
    }, { status: 500 });
  }
}
```

Visit `http://localhost:3000/api/test-email` to test your setup.

🔧 Configuration Details

Required Environment Variables

Variable	Description	Example
`GMAIL_CLIENT_ID`	OAuth2 Client ID from Google Cloud Console	`123...googleusercontent.com`
`GMAIL_CLIENT_SECRET`	OAuth2 Client Secret	`GOCSPX-...`
`GMAIL_REFRESH_TOKEN`	Refresh token from OAuth Playground	`1//04...`
`GMAIL_USER_EMAIL`	Your Gmail address for sending emails	`your-email@gmail.com`
`GMAIL_REDIRECT_URL`	OAuth redirect URL (optional)	`https://developers.google.com/oauthplayground`

Security Best Practices

1. **Keep credentials secure**: Never commit `.env.local` to version control
2. **Use least privilege**: Only grant necessary Gmail scopes
3. **Rotate tokens**: Periodically regenerate your credentials
4. **Monitor usage**: Check Google Cloud Console for API usage

🚨 Troubleshooting

Common Issues

"Invalid grant" error

- **Cause**: Refresh token expired or invalid

- **Solution**: Generate a new refresh token using OAuth Playground

"Access denied" error

- **Cause**: Gmail API not enabled or insufficient permissions
- **Solution**: Ensure Gmail API is enabled and correct scopes are granted

"Quota exceeded" error

- **Cause**: Gmail API quota limits reached
- **Solution**: Check quotas in Google Cloud Console or request quota increase

"Authentication error"

- **Cause**: Wrong Client ID/Secret or missing environment variables
- **Solution**: Double-check all environment variables match your Google Cloud credentials

Debug Steps

1. **Check environment variables**:

   ```
   echo $GMAIL_CLIENT_ID
   echo $GMAIL_CLIENT_SECRET
   echo $GMAIL_REFRESH_TOKEN
   echo $GMAIL_USER_EMAIL
   ```

2. **Test API connection**: Visit `/api/test-email` endpoint

3. **Check server logs**: Look for Gmail API error messages in your Next.js console

4. **Verify Google Cloud setup**:

   - Gmail API is enabled
   - OAuth consent screen is configured
   - Correct redirect URI is set

📊 Usage Limits

- **Daily quota**: 1 billion requests per day (default)
- **Per-user rate limit**: 250 quota units per user per 100 seconds
- **Sending limits**: Standard Gmail sending limits apply

🔄 Production Deployment

For production deployment:

1. **Update OAuth consent screen** to "In production"
2. **Add your domain** to authorized domains
3. **Set environment variables** in your hosting platform

4. **Monitor email delivery** and API usage

📧 Features Available

Your ROI Calculator now supports:

- ☑ **ROI report emails** - Detailed PDF-style reports sent to users
- ☑ **Admin notifications** - Lead alerts sent to your team
- ☑ **Confirmation emails** - Thank you messages to prospects
- ☑ **Professional formatting** - HTML emails with your branding
- ☑ **Reply-to handling** - Proper email threading
- ☑ **Error handling** - Graceful fallbacks if email fails

🎉 You're Ready!

Your Gmail API is now configured and ready to send professional ROI calculation reports to your prospects!

Need help? Check the Gmail API documentation or contact support.