

Google Photos/Drive Integration - Report

1. Features Added

A. UI/UX Enhancements – Generate Narrative Button

- Implemented a robust loading-state mechanism to prevent duplicate API calls.
- Added `isGenerating` state in `CreateEntry.js` with:
 - Disabled button during processing
 - CSS spinner animation
 - Dynamic button label: “**Generating...**”
- Updated `Compose.js` to follow the same pattern for a consistent UX.

B. Google Photos / Google Drive Integration

- Added “**Import from Google Photos**” entry point in `CreateEntry.js`.
- Built `PhotosPicker.js` to display fetched Google Drive/Photos thumbnails.
- Implemented backend endpoints for OAuth + image retrieval:
 - `/photos/auth` — starts incremental OAuth with `drive.readonly`
 - `/api/photos/recent` — fetches recent images from Google Drive
 - `/google/drive/images` — alternative Drive image listing endpoint
- Implemented mature token-handling pipeline:
 - Incremental authorization prevents consent-loop issues
 - Token merging logic in `/google/callback`
 - Session-based storage with cookie credentials
 - `include_granted_scopes: true` to reduce repeated prompts

2. Problems Encountered

A. API Access Failures

- Google Photos Library API returned repeated:
 - 403: SERVICE_DISABLED
 - "API not enabled for project 118341865906"
- The OAuth credentials belonged to a Google Cloud project where:
 - Drive API = disabled
 - Photos Library API = disabled
- Attempts to access Drive/Photos APIs resulted in:
 - Permission errors
 - API quota warnings
 - Authentication failures

B. Authentication & Token Management Complexities

- Refresh tokens expired silently, causing recurring 401/403 loops.
- Needed logic to verify:
 - Valid token
 - Valid scope
 - Valid session
- Test API calls failed because APIs weren't enabled on the GCP side.

C. User-Facing Application Issues

- End users could not access Google Photos import at all:
 - App still in **Google OAuth “Testing” Mode**
 - Only developer-approved test emails allowed
 - Google verification not completed
- This caused the 403:

"AIvisionstoryjournal has not completed the Google verification process. Access denied."

3. Root Causes

A. APIs Disabled in Google Cloud Project

The Google Cloud project (118341865906) tied to the OAuth credentials had:

- **Google Drive API – Disabled**
- **Google Photos Library API – Disabled**

Since Google requires explicit permission per project, every request failed with:

SERVICE_DISABLED: API not enabled

B. Missing Permissions to View/Modify the GCP Project

The account attempting to fix the issue did NOT have `resourcemanager.projects.get` → meaning the developer couldn't even open the project in Google Cloud Console.

C. OAuth Consent Screen Not Verified

The application is in "Testing" mode:

- Limited to **100 test users**
- Only emails manually added by the developer can log in
- Regular users get blocked with 403 “access denied”

4. How the Issues Were Resolved

Step 1 – Enable Required Google APIs

Once given access by the project owner:

- **Google Drive API** enabled
- **Google Photos Library API** enabled

This removed the SERVICE_DISABLED barrier.

Step 2 – Update OAuth & Token Handling

- Cleaned the OAuth flow
- Added token merge logic
- Removed old expired test tokens
- Rebuilt incremental authorization to ensure scope alignment

Step 3 – Test the Integration End-to-End

- Google Drive endpoints began returning valid thumbnails
 - The frontend photo picker successfully rendered images
 - Multi-click issues fixed via improved loading states
-

5. Why End Users Still Cannot Access Google Photos Import

Regular users are blocked NOT because of their Google accounts, but because *your app is not verified by Google yet.*

Verification Status

- App = *In Google OAuth Testing Mode*
- Only approved test users may authenticate
- Anyone else gets:

Error 403: access denied

Why Google blocks this

Google protects high-risk scopes like:

- `photoslibrary.readonly`
- `drive.readonly`

Any app using them must undergo verification to prevent:

- Abuse
- Misleading apps
- Unauthorized storage access

Until verification is completed:

- Only whitelisted test users can access Google Photos integration.
- All other end users get blocked by Google automatically.

6. Final Summary

This integration required multiple layers of fixes:

What Was Built

- Stronger loading state & UX
- OAuth with incremental scope granting
- Drive/Photos endpoints
- Photo picker UI

What Broke

- API disabled → 403 SERVICE_DISABLED
- Missing GCP permissions → couldn't inspect project
- App unverified → end users blocked

How It Was Fixed

- Enabled required Drive/Photos APIs
- Updated OAuth flow and token model
- Cleaned up error handling
- Gained correct GCP project access

Remaining Limitation

End users cannot use Google Photos yet because the app remains in Google OAuth “Testing Mode,” and Google requires a full verification process before making it public.