

# Claude Code Prompts for Debugging Listing Lift Photo Upload Issues

## Context

This is a Next.js 14 app that allows homeowners to upload photos for AI enhancement. Photos are uploaded to AWS S3 using presigned URLs. The app uses Prisma for database and stores the S3 cloud\_storage\_path in the database.

## Known Issues

1. **HEIC photo previews don't work during upload** - JPEG, PNG, WebP work fine, but HEIC files show broken previews
2. **Admin dashboard shows broken photos** - When admin views submissions, all photos appear broken/can't load, suggesting S3 upload may be failing

## PROMPT 1: Diagnose the Photo Upload Flow

I have a Next.js photo upload app with the following issue: photos appear broken **in** the admin dashboard after being submitted. Here's **the upload flow**:

1. User selects photos **in** PhotoUploader component
2. On submit, SubmissionForm calls uploadPhotos() which:
  - Gets presigned URL from /api/upload/presigned
  - Uploads file directly to S3 using PUT request
  - Stores the **cloud\_storage\_path** **in** database
3. Admin views photo by fetching URL from /api/file-url

Please analyze the following code **and** identify potential issues **in** the upload flow:

[PASTE THE CODE FROM ALL\_CODE.md HERE]

Specifically check:

1. Is the presigned URL being generated correctly?
2. Is the file being uploaded to S3 with correct headers?
3. Is the Content-Disposition header being handled properly?
4. Is the **cloud\_storage\_path** being stored correctly **in** the database?
5. Is the file URL being retrieved correctly **for** display?
6. Could there be CORS **or** permission issues?

## PROMPT 2: Debug HEIC Preview Specifically

I'm using heic2any library to convert HEIC files to JPEG for preview, but HEIC previews are still broken. The code is in photo-uploader.tsx.

Here's the relevant code:

[PASTE photo-uploader.tsx CODE HERE]

The issue:

- JPEG, PNG, WebP previews work fine
- HEIC files show broken preview thumbnails
- The heic2any conversion **is** supposed to happen **in** convertHeicToJpeg function

Please analyze:

1. Is heic2any being imported **and** used correctly?
2. Is the async/await flow correct **for** HEIC conversion?
3. Could there be a race condition **or** timing issue?
4. Is the blob URL being created properly after conversion?
5. Are there any error handling gaps?

## PROMPT 3: Debug S3 Upload Specifically

Photos uploaded to S3 via presigned URLs appear broken when retrieved. Here's the flow:

\*\*Presigned URL Generation (lib/s3.ts):\*\*  
 [PASTE s3.ts CODE HERE]

\*\*Upload API Route (app/api/upload/presigned/route.ts):\*\*  
 [PASTE upload-presigned-route.ts CODE HERE]

\*\*File URL Retrieval (app/api/file-url/route.ts):\*\*  
 [PASTE file-url-route.ts CODE HERE]

\*\*Client Upload Code (submission-form.tsx - uploadPhotos function):\*\*  
 [PASTE the uploadPhotos function from submission-form.tsx]

Please analyze:

1. Is the presigned URL being generated with correct parameters?
2. Is Content-Disposition being set correctly **for** the signed headers?
3. When client uploads, are the headers matching what the presigned URL expects?
4. Is the public/private path logic correct?
5. Is getFileUrl returning the correct URL format?
6. Could the region **or** bucket configuration be wrong?

## PROMPT 4: Full Diagnostic Request

I have a Next.js 14 + Prisma + AWS S3 photo upload app with two issues:

\*\*Issue 1: HEIC Preview Broken\*\*

When users upload HEIC photos, the preview thumbnails are broken. Other formats (JPEG, PNG, WebP) work fine.

\*\*Issue 2: Photos Broken **in** Admin Dashboard\*\*

After submission, when admin views the photos, they all appear **as** broken images (see a attached screenshot showing "234" text with broken image icon).

\*\*Architecture:\*\*

- Photos uploaded via presigned URLs to S3
- cloud\_storage\_path stored **in** Prisma/PostgreSQL database
- Admin fetches file URL via /api/file-url endpoint
- Using AWS SDK v3

Please analyze ALL the following code files **and** provide:

1. Root cause analysis **for** each issue
2. Specific code changes needed to fix each issue
3. Any additional debugging steps I should take

[PASTE ALL CODE FROM ALL\_CODE.md HERE]

## PROMPT 5: Quick Check for Common Issues

Quick diagnostic check **for** S3 presigned URL upload issues:

1. In this code, when isPublic=**true** **and** ContentDisposition **is** set, does the presigned URL include content-disposition **in** signed headers?
2. If yes, **is** the client sending the Content-Disposition header when uploading?
3. Check this upload code - **is** it sending the right headers?

```
typescript
// From submission-form.tsx
const hasContentDisposition = uploadUrl?.includes?('content-disposition');
const uploadHeaders: Record<string, string> = {
  'Content-Type': contentType
};
if (hasContentDisposition) {
  uploadHeaders['Content-Disposition'] = 'attachment';
}

const uploadRes = await fetch(uploadUrl, {
  method: 'PUT',
  body: photo.file,
  headers: uploadHeaders
});
```

Is there a mismatch between what the presigned URL expects and what the client is sending?

...

## What to Bring Back to DeepAgent

---

After Claude Code analyzes the issues, please bring back:

1. The specific root cause(s) identified
2. The exact code changes recommended
3. Any additional context or questions from Claude Code

I can then implement the fixes based on Claude Code's analysis.