Google Cloud Storage Setup for Cloud Run - Complete GUI Guide

📋 Overview

This guide will walk you through setting up persistent file storage for your Hotel Shift Log application using Google Cloud Storage (GCS) mounted as a volume. **No code changes required!**

What you'll do:
1. Create a storage bucket (3 minutes)
2. Grant permissions (2 minutes)
3. Mount the bucket to Cloud Run (5 minutes)
4. Verify it works (5 minutes)

Total time: ~15-30 minutes

✅ What I Changed in Your Code

Answer: NOTHING! 🎉

Your existing code already works perfectly with GCS FUSE volumes because:
- It uses standard filesystem operations (`fs.writeFile` , `fs.readFile`)
- It stores files in `/uploads/` directory
- When we mount GCS as `/uploads/` , your code automatically uses cloud storage

Files checked:
- ✅ `app/api/files/[...filename]/route.ts` - File serving
- ✅ `app/api/comments/route.ts` - Comment attachments
- ✅ `app/api/reports/create/route.ts` - Report attachments

All use `path.join(process.cwd(), 'uploads')` which will automatically point to the mounted GCS bucket.

🚀 Step-by-Step GUI Instructions

Step 1: Create a Google Cloud Storage Bucket

Navigate to: Google Cloud Storage Console (https://console.cloud.google.com/storage/browser)

1. **Click "CREATE BUCKET"** (blue button at the top)

2. **Name your bucket:**
   `hotel-shift-log-uploads`
   ⚠️ **Important:** Bucket names must be globally unique. If this is taken, try:
   - `hotel-shift-log-uploads-[your-hotel-name]`

- `hotel-shift-log-uploads-[random-number]`
- Example: `hotel-shift-log-uploads-12345`

3. **Choose where to store your data:**
   - Select **“Region”**
   - Choose the **same region as your Cloud Run service** (example: `us-central1 (Iowa)`)
   - Why? Lower latency and no data transfer costs
4. **Choose a default storage class:**
   - Select **“Standard”**
   - This is best for frequently accessed files
5. **Choose how to control access:**
   - Select **“Uniform”** (recommended)
   - This simplifies permissions
6. **Choose how to protect object data:**
   - Leave defaults:
     - ✅ None (no data protection tool)
     - Soft delete policy: 7 days (default)
     - You can enable versioning later if needed
7. **Click “CREATE”**

✅ **Checkpoint:** You should see your new bucket in the list!

Step 2: Grant Permissions to Cloud Run Service Account

Your Cloud Run service needs permission to read/write files in the bucket.

Navigate to: Your bucket → Click on the bucket name `hotel-shift-log-uploads`

1. **Click the “PERMISSIONS” tab** (at the top of the page)
2. **Click “+ GRANT ACCESS”** button
3. **Add new principals:**
   - In the “New principals” field, enter:
   `143559442445-compute@developer.gserviceaccount.com`
   - ⚠️ **Replace** `143559442445` with YOUR project number if different
   - To find your project number:
     - Click the project dropdown at the very top
     - Your project number is shown next to the project name
4. **Select a role:**
   - In the “Select a role” dropdown, search for:
   `Storage Object Admin`
   - Select **“Storage Object Admin”**
   - This allows read/write/delete operations
5. **Click “SAVE”**

✅ **Checkpoint:** You should see the service account listed in the permissions with “Storage Object Admin” role!

Step 3: Mount the Bucket to Cloud Run

Now we’ll configure Cloud Run to mount this bucket as the `/uploads` directory.

Navigate to: Cloud Run Console (https://console.cloud.google.com/run)

1. **Click on your service:** `hotel-shift-log`
2. **Click “EDIT & DEPLOY NEW REVISION”** (button at the top)
3. **Scroll down to “Container(s), Volumes, Networking, Security”** section
   - Click to expand if collapsed
4. **Click the “VOLUMES” tab**
5. **Click “+ ADD VOLUME”** button
6. **Configure the volume:**

Volume type: Select **“Cloud Storage bucket”**

Name: Enter a name for this volume:
`uploads`

Bucket: Select your bucket from the dropdown:
`hotel-shift-log-uploads`
(Or whatever name you used if the above was taken)

Mount options: Leave blank (defaults are fine)

☑️ **Mount as read-only:** Leave **UNCHECKED** (app needs to write files)

1. **Click “MOUNT”** button at the bottom of the volume configuration
2. **Now configure the mount path:**
   - After clicking Mount, you’ll see a “VOLUME MOUNTS” section appear
   - Look for your `uploads` volume
   - In the **“Mount path”** field, enter:
   `/app/nextjs_space/uploads`
   - ⚠️ **Important:** This exact path is where your app expects the uploads folder
3. **Verify your configuration:**
   - Volume name: `uploads`
   - Bucket: `hotel-shift-log-uploads` (or your bucket name)
   - Mount path: `/app/nextjs_space/uploads`
   - Read-only: **OFF** (unchecked)
4. **Enable Cloud Run Gen2 Execution Environment:**
   - Scroll to the **“CONTAINER” tab**
   - At the bottom, look for **“Execution environment”**
   - Select **“Second generation”**

- ⚠️ **Critical:** Volumes only work with 2nd gen!

5. **Click "DEPLOY"** at the bottom

Wait for deployment... (This will take 3-5 minutes)

✅ **Checkpoint:** Once deployed, you should see "Service deployed successfully" message!

Step 4: Verify the Setup

Let's make sure files are actually being stored in GCS!

Test 1: Upload a File via Your App

1. **Open your deployed app** (Cloud Run URL)
2. **Log in** as an employee (username: `employee`, password: use the one you set)
3. **Create a new shift report:**
   - Add some text
   - **Attach a file** (any image or document)
   - Submit the report
4. **Check if file is in GCS:**
   - Go back to Cloud Storage Console (https://console.cloud.google.com/storage/browser)
   - Click on your bucket `hotel-shift-log-uploads`
   - You should see a file with a random name like `1761234567890-abc123.jpg`
   - ✅ **Success!** Files are now stored in cloud storage

Test 2: Add a Comment with Attachment

1. **Log in as a manager** (username: `manager`)
2. **Open the report** you just created
3. **Add a comment with an attachment:**
   - Upload an image
   - Submit the comment
4. **Check GCS:**
   - Go to your bucket → `comments/` folder
   - You should see the comment attachment
   - ✅ **Success!** Comment files also stored in cloud

Test 3: Download Files (Verify Access)

1. **In your app,** click to view/download the attachments you uploaded
2. **Files should display correctly**
   - Images should show previews
   - Documents should download
   - ✅ **Success!** File serving works

Test 4: Redeploy (Persistence Test)

1. **Make a small code change** (or just redeploy):
   - Go to Cloud Run → Edit & Deploy New Revision

- Don’t change anything
- Just click DEPLOY

2. **After redeployment:**
   - Open your app
   - Navigate to the reports you created
   - ✅ **Success!** Attachments are still there!
   - 🎉 **Files survived the redeploy!**

🔍 Troubleshooting

Problem: “Volume not mounting” or “Permission denied”

Solution:
1. Verify the service account has “Storage Object Admin” role on the bucket
2. Check that you’re using **2nd generation execution environment**
3. Verify mount path is exactly: `/app/nextjs_space/uploads`

To check execution environment:
- Cloud Run → Your Service → Edit & Deploy
- Container tab → scroll to bottom → Execution environment
- Should say “Second generation”

Problem: “Files not appearing in GCS”

Solution:
1. Check Cloud Run logs for errors:
- Cloud Run → Your Service → LOGS tab
- Look for file upload errors
2. Verify mount path matches app’s upload directory
3. Check that bucket permissions are correct

To view logs:

```
Cloud Run Console → hotel-shift-log → LOGS tab
Filter: "upload" or "file"
```

Problem: “Bucket name already taken”

Solution:
1. Choose a different bucket name (must be globally unique)
2. Update the mount configuration to use your new bucket name
3. Example names:
- `hotel-shift-log-uploads-yourhotelname`
- `hotel-shift-log-uploads-12345`
- `shift-reports-production-storage`

Problem: “Old files from local storage are missing”

Expected behavior: Files stored before GCS setup are on the old container’s local disk and are not automatically migrated.

Solution (if you need old files):
1. Before setting up GCS, manually download the old uploads directory
2. Upload files to GCS bucket manually via console
3. Or accept that only new files will be persisted

Note: Old files were going to be lost anyway on next redeploy!

📊 Monitoring Your Storage

View Storage Usage

Navigate to: Cloud Storage Console (https://console.cloud.google.com/storage/browser) → Your bucket

You can see:
- Total objects (number of files)
- Total size (GB)
- Storage class distribution

View Costs

Navigate to: Cloud Billing Reports (https://console.cloud.google.com/billing/reports)

Filter by:
- Service: “Cloud Storage”
- SKU: “Standard Storage”

Expected costs:
- ~$0.02 per GB per month
- Example: 10GB = $0.20/month

🎯 What You’ve Accomplished

✅ **Persistent file storage** - Files never lost on redeploy
✅ **Zero code changes** - Used existing filesystem code
✅ **Automatic backups** - GCS is durable and replicated
✅ **Scalable** - Works with multiple Cloud Run instances
✅ **Cost-effective** - Pennies per month for typical usage
✅ **Production-ready** - Industry standard approach

📝 Configuration Summary

For your records:

Setting	Value
GCS Bucket Name	`hotel-shift-log-uploads` (or yours)
Region	`us-central1` (same as Cloud Run)
Storage Class	Standard
Access Control	Uniform
Service Account	`143559442445-com-pute@developer.gserviceaccount.com`
Permissions	Storage Object Admin
Volume Name	`uploads`
Mount Path	`/app/nextjs_space/uploads`
Read-only	No
Execution Environment	Gen 2 (2nd generation)

🔄 Future Maintenance

Backing Up Files

GCS already provides durability, but if you want versioning:

1. Go to your bucket → **Configuration** tab
2. Enable **Object Versioning**
3. Old versions of files will be kept even if overwritten

Cleaning Up Old Files (Optional)

If you want to automatically delete old files after a certain period:

1. Go to your bucket → **Lifecycle** tab
2. Click **ADD A RULE**
3. Example: Delete files older than 365 days
4. Action: “Delete object”
5. Condition: Age > 365 days

Monitoring

Set up alerts for:
- Storage usage approaching limits
- Unexpected costs
- High error rates

Navigate to: Cloud Monitoring (https://console.cloud.google.com/monitoring)

🎉 You’re Done!

Your Hotel Shift Log application now has **production-grade file storage** that:
- ✅ Persists across redeployments
- ✅ Works with zero code changes
- ✅ Costs pennies per month
- ✅ Scales automatically
- ✅ Is backed by Google’s infrastructure

Next deployment: Just push code to GitHub - files will stay safe in GCS! 🚀

Document created: October 23, 2025
Last updated: October 23, 2025