



# Fix: “Bucket not found” Error

## Problem

You’re seeing this error when uploading logos:


```
Failed to upload file: Bucket not found
```

## Root Cause

The Supabase Storage bucket `customer-logos` doesn’t exist in your Supabase project yet.

## Solution (5 minutes)

### Step 1: Go to Supabase Storage Dashboard



1. Open <https://supabase.com/dashboard>
2. **Select your project** (the one connected to your app)
3. In the left sidebar, click **Storage** ( icon)

### Step 2: Create the Bucket

1. Click the **“New bucket”** button (top right)
2. Fill in the form:
  - **Name:** `customer-logos` (exactly this, no spaces or typos!)
  - **Public bucket:** ☒ **Enable this checkbox** (IMPORTANT!)
  - **File size limit:** Leave default (50MB is fine)
  - **Allowed MIME types:** Leave empty (allows all image types)
3. Click **“Create bucket”**

### Step 3: Verify the Bucket

You should now see `customer-logos` in your buckets list with:

-  **Public** badge (if not, see Step 4)
-  0 objects

### Step 4: Set Public Access (if needed)

If you forgot to enable “Public bucket” during creation:

1. Click on the `customer-logos` bucket
2. Go to **Configuration** tab
3. Toggle **“Public bucket”** to ON
4. Click **Save**

### Step 5: Set Storage Policies (Optional - for extra security)

This step is optional but recommended. It adds proper access rules:

1. In Supabase Dashboard, go to **SQL Editor** (left sidebar)

2. Click **"New query"**
3. Paste this SQL and click **Run**:

```
-- Allow public read access to customer logos
CREATE POLICY "Public Access"
ON storage.objects FOR SELECT
USING (bucket_id = 'customer-logos');

-- Allow authenticated uploads
CREATE POLICY "Authenticated uploads"
ON storage.objects FOR INSERT
TO authenticated
WITH CHECK (bucket_id = 'customer-logos');

-- Allow authenticated updates
CREATE POLICY "Authenticated updates"
ON storage.objects FOR UPDATE
TO authenticated
USING (bucket_id = 'customer-logos');

-- Allow authenticated deletes
CREATE POLICY "Authenticated deletes"
ON storage.objects FOR DELETE
TO authenticated
USING (bucket_id = 'customer-logos');
```

## Step 6: Test Logo Upload

1. Go back to your app: <https://roblox-tool-ruddy.vercel.app/admin>
2. Navigate to **Customers** tab
3. Click **Logo** button for any customer
4. Try uploading an image
5. ☒ You should see: **"Logo uploaded successfully to cloud storage"**

## Quick Checklist

- ☐ Bucket named exactly `customer-logos` (no typos!)
- ☐ Bucket is marked as **Public**
- ☐ You can see the bucket in Storage dashboard
- ☐ Policies created (optional but recommended)
- ☐ Logo upload test successful


## Visual Guide

### What Your Storage Dashboard Should Look Like:

```
Storage
├──  customer-logos (Public) ← You should see this!
│   └── (empty for now)
└──
```

### Bucket Settings:

- **Name:** customer-logos

- **Status:**  Public
- **Size limit:** 50 MB (default)
- **Files:** 0 (until you upload)

## Common Mistakes

- ✗ **Wrong bucket name** → Must be exactly `customer-logos` (with dash, lowercase)
- ✗ **Private bucket** → Must be PUBLIC for the app to access uploaded images
- ✗ **Wrong Supabase project** → Make sure you're in the same project as your DATABASE\_URL

## How to Verify Your Supabase Project

The bucket must be created in the **same Supabase project** that your app uses.

To verify you're in the correct project:






1. Check your DATABASE\_URL in Vercel → Settings → Environment Variables
2. Look for the project ID (e.g., `rxyhdfellmsssgsfci` in your connection string)
3. In Supabase Dashboard, click your project name (top left)
4. The project ID should match!

Example DATABASE\_URL format:

```
postgresql://postgres:[PASSWORD]@db.[PROJECT_ID].supabase.co:5432/postgres
                        ^^^^^^^^^^^^^^^^^
                        This is your project ID
```

## Success Indicators

After completing these steps, you should see:

1.  Bucket appears in Storage dashboard
2.  Bucket has "Public" badge
3.  Logo upload works without errors
4.  Uploaded logos display correctly
5.  Logos persist after deployment

## What Happens After You Create the Bucket?

Once the bucket exists:

- Your app can upload customer logos
- Logos are stored in the cloud (Supabase Storage)
- Each customer can have their own custom logo
- Logos persist across deployments
- Public URLs are generated for each logo

## Why Did This Happen?

The code **tries** to auto-create the bucket, but sometimes this fails due to:

- Insufficient permissions

- Supabase API rate limits
- Network issues during deployment

**Manual creation** is the most reliable method!



---

After creating the bucket, your logo upload feature should work perfectly!

Test it by uploading a logo for one of your customers.

---

**Still having issues?** Check the full setup guide: `SUPABASE_STORAGE_SETUP.md`