

Yes — if you're stuck in a CORS error loop after explicitly setting CORS permissions in `app.json`, it likely means **either the CORS policy is not being properly registered** by the server, **or the frontend and backend domains are mismatched or out of sync**. Here's a focused checklist and some actionable suggestions to break out of the loop:

✓ Immediate Debug Checklist

1. Verify CORS is correctly defined

If you're using something like Express or Flask for the backend, ensure:

```
// app.json or your CORS middleware config
{
  "server": {
    "cors": {
      "origin": ["https://app.tempo.build"],
      "methods": ["GET", "POST", "PUT", "DELETE", "OPTIONS"],
      "credentials": true
    }
  }
}
```

⚠ Some platforms require `"origin": "*" not to be used with credentials: true`. If using credentials, specify the exact origin.

2. Tempo expects headers to be returned consistently

Tempo will reject your dev server if your CORS headers:

- Are missing on `OPTIONS` (preflight) requests
- Are not symmetrical with actual `GET/POST` requests

✓ Ensure **all responses**, including preflight (`OPTIONS`), include:

Access-Control-Allow-Origin: `https://app.tempo.build`

Access-Control-Allow-Credentials: true
Access-Control-Allow-Headers: Content-Type, Authorization

3. Dev Server Restart May Not Apply Changes

Some dev environments (like Vite, Webpack, or Next.js) don't fully apply CORS config until a hard stop/start (not hot reload).

Solution:

- Fully stop the dev server process (not just reload)
- Restart clean: `npm run dev` or `yarn dev`

Also clear any cache in the browser or try an incognito window.

If You're Still Stuck

4. Confirm Dev Server Base URL Matches in Tempo App Config

In your Tempo project, make sure the local dev server **exactly matches** what the frontend tries to connect to:

```
"proxy": "http://localhost:3000"
```

But if Tempo routes requests to a secure tunnel (like `https://localhost.tempo.dev`), your CORS config needs to match **that** domain.

5. Add Debug Headers Temporarily

Add a route that echoes headers:

```
app.get("/debug-headers", (req, res) => {  
  res.set("Access-Control-Allow-Origin", "https://app.tempo.build");  
  res.set("Access-Control-Allow-Credentials", "true");  
  res.set("Access-Control-Allow-Headers", "Content-Type, Authorization");  
});
```

```
res.send("CORS headers sent");
});
```

Hit it from the browser or Tempo's dev console to confirm headers are received.

Suggested Fix Summary

Issue	Fix
CORS headers not returned on <code>OPTIONS</code>	Add CORS middleware early in server
Wrong origin in header	Use <code>https://app.tempo.build</code> exactly
<code>credentials: true</code> conflicts	Avoid using "*" with credentials
Dev server restart ineffective	Kill process and relaunch fully
Tempo blocking silently	Try hard refresh or open dev tools > Network

If you want to paste your `app.json`, server code snippet, or exact error message, I can offer more direct help. Would you like to do that?