



Sentry Debug Deployment Checklist

Date: January 2, 2026

Commit: 85ac03f - "debug: Add comprehensive Sentry debugging infrastructure"

Status: Ready to Push and Deploy



CRITICAL: The Likely Issue

Based on the log message [Sentry] Not running in browser environment , the client-side Sentry is not initializing because:

`NEXT_PUBLIC_SENTRY_DSN` is NOT set in Render environment variables

This is why you see:

- Server-side: "initialization successful"
 - Edge: "initialization successful"
 - Client-side: "Not running in browser environment"
-



Step-by-Step Deployment



STEP 1: Push to GitHub

```
cd /home/ubuntu/carelinkai-project
git push origin main
```

Expected Result: Commit 85ac03f pushed successfully



STEP 2: Get Your Sentry DSN

1. Log into [Sentry](https://sentry.io) (<https://sentry.io>)
2. Go to **Settings** → **Projects**
3. Select your project
4. Go to **Client Keys (DSN)**
5. Copy the DSN (looks like: `https://abc123@o123456.ingest.sentry.io/123456`)

Keep this DSN handy for the next step!



STEP 3: Add Environment Variable to Render

1. Go to [Render Dashboard](https://dashboard.render.com) (<https://dashboard.render.com>)
2. Select your **carelinkai-project** service
3. Click **Environment** in the left sidebar

4. Click **Add Environment Variable**

Add these variables:

```
# PRIMARY (CRITICAL for client-side)
Key:   NEXT_PUBLIC_SENTRY_DSN
Value: <paste-your-sentry-dsn-here>

# OPTIONAL (server/edge likely already works, but good to have)
Key:   SENTRY_DSN
Value: <paste-your-sentry-dsn-here>
```

1. Click **Save Changes**

✓ **STEP 4: Deploy with Build Cache Clear**

⚠ **CRITICAL:** You MUST clear the build cache!

Why? Next.js bundles `NEXT_PUBLIC_` environment variables at BUILD time, not runtime. A simple restart will NOT work.

How to Deploy:

1. In Render, go to **Manual Deploy**
2. ✓ **Check the box:** “Clear build cache”
3. Click **“Deploy latest commit”**

Expected Timeline:

- Build: ~5-10 minutes
- Deploy: ~2-3 minutes
- Total: ~15 minutes

✓ **STEP 5: Monitor Deployment Logs**

What to Watch For:

Open the Render logs and look for these debug messages during startup:

✓ **GOOD: Client Config Success**

```
[Sentry Debug] ===== CLIENT CONFIG =====
[Sentry Debug] Browser Environment: true
[Sentry Debug] NEXT_PUBLIC_SENTRY_DSN exists: true 🟢 LOOK FOR THIS!
[Sentry Debug] NEXT_PUBLIC_SENTRY_DSN length: 89
[Sentry Debug] NODE_ENV: production
[Sentry] ✓ Client-side initialization successful
[Sentry Debug] Sentry.isInitialized(): true
```

✗ **BAD: Still Missing (If you see this, env var not set correctly)**

```
[Sentry Debug] NEXT_PUBLIC_SENTRY_DSN exists: false 🟠 PROBLEM!
[Sentry] ✗ CRITICAL: NEXT_PUBLIC_SENTRY_DSN is not set
```

✓ GOOD: Server Config Success

```
[Sentry Debug] ===== SERVER CONFIG =====
[Sentry Debug] SENTRY_DSN exists: true
[Sentry Debug] Using DSN: FOUND
[Sentry] ✓ Server-side initialization successful
```

✓ GOOD: Edge Config Success

```
[Sentry Debug] ===== EDGE CONFIG =====
[Sentry Debug] SENTRY_DSN exists: true
[Sentry] ✓ Edge initialization successful
```

All three should show ✓ success!

✓ STEP 6: Test Server-Side Sentry

Once deployed, test the server endpoint:

```
# Replace with your actual Render URL
curl https://carelinkai.onrender.com/api/test-sentry-server
```

Expected Response:

```
{
  "success": true,
  "message": "Server-side Sentry tests completed",
  "details": {
    "sentryInitialized": true,
    "messageId": "abc123...",
    "errorId": "def456...",
    "timestamp": "2026-01-02T..."
  }
}
```

Check Render Logs for:

```
[Sentry Test] All server-side tests completed!
[Sentry Debug] Server beforeSend called - event will be sent to Sentry
[Sentry Test] Events flushed to Sentry
```

✓ STEP 7: Test Edge Runtime Sentry

```
curl https://carelinkai.onrender.com/api/test-sentry-edge
```

Expected Response: Similar to server test, but with `"runtime": "edge"`




✅ STEP 8: Test Client-Side Sentry

Open in your browser:

```
https://carelinkai.onrender.com/test-sentry-client
```




What You Should See:

1. Status Section:

-  Sentry Initialized: **Yes** 
- Environment: **production**
- NEXT_PUBLIC_SENTRY_DSN:  **Set**

2. No Warning Box: (If env var is set correctly, the red warning won't appear)

3. Test Buttons:

- Click “ Test Message”
- Click “ Test Error”
- Click “ Test With Context”

4. Activity Log:

- Should show success messages
- Should show event IDs


5. Browser Console (F12):

```
[Sentry Test Client] Message ID: abc123...
```










```
[Sentry Test Client] Error ID: def456...
```

If you see debug mode logs in the console, that's good! It means Sentry is working.

✅ STEP 9: Verify in Sentry Dashboard

1. Log into [Sentry](https://sentry.io) (<https://sentry.io>)
2. Go to **Issues** → **All Issues**
3. Look for events with  **emoji** in the title

You should see (within 1-2 minutes):


Event Title	Type	Source
 Server-side Sentry test error	Error	Server
 Server-side Sentry test message	Message	Server
 Server-side test with context	Message	Server
 Edge runtime Sentry test error	Error	Edge
 Edge runtime Sentry test message	Message	Edge
 Edge runtime test with context	Message	Edge
 Client-side Sentry test error	Error	Browser
 Client-side Sentry test message	Message	Browser
 Client-side test with context	Message	Browser

Total: 9 test events (if you clicked all test buttons)

Click on any event to verify:

- ☒ Breadcrumbs are attached
- ☒ Context/tags are present
- ☒ Stack trace is available
- ☒ Environment is set correctly

STEP 10: Test the Floating Debug Button

1. Go to operator dashboard: `https://carelinkai.onrender.com/operator`
2. Look for **purple  button** in bottom-right corner
3. Click it to open debug menu
4. Try the quick test buttons

Note: Button only appears if:

- You're in development, OR
- `NEXT_PUBLIC_SHOW_DEBUG_TOOLS=true` is set in Render

To enable in production (optional):

```
# Add to Render environment variables
NEXT_PUBLIC_SHOW_DEBUG_TOOLS=true
```

✅ STEP 11: Verify Real Error Tracking

Create a real test error to confirm production errors are tracked:

1. Visit any page in your app
2. Open browser console (F12)
3. Type and run:

```
javascript
  throw new Error("Real production test error");
```

4. Check Sentry dashboard in 1-2 minutes

You should see: A new error event in Sentry (without 🐛 emoji)

🎯 Success Criteria

All Green Checks = Success ✅

- [] Commit pushed to GitHub
- [] NEXT_PUBLIC_SENTRY_DSN added to Render
- [] Deployed with build cache cleared
- [] Logs show: "NEXT_PUBLIC_SENTRY_DSN exists: true"
- [] Logs show: All three configs initialized successfully
- [] /api/test-sentry-server returns success
- [] /api/test-sentry-edge returns success
- [] /test-sentry-client shows "Initialized: Yes ✅"
- [] Test events appear in Sentry dashboard
- [] Events include breadcrumbs and context
- [] Real errors are being captured

🐛 Troubleshooting

Issue: Still shows "NEXT_PUBLIC_SENTRY_DSN exists: false"

Solutions:

1. Double-check environment variable is spelled correctly in Render
2. Make sure you cleared build cache when deploying
3. Try manual deploy again with cache clear
4. Verify DSN value doesn't have extra spaces or quotes

Issue: "NEXT_PUBLIC_SENTRY_DSN exists: true" but no events

Solutions:

1. Check DSN is correct (copy from Sentry dashboard again)

2. Check Sentry project status (not disabled/paused)
3. Check Sentry quotas (not exceeded)
4. Look for `beforeSend` filter messages in logs

Issue: Test endpoints return 404

Solutions:

1. Wait for deployment to complete
2. Check build logs for errors
3. Verify routes were created in `src/app/api/`

Issue: Can't see debug button

Solutions:

1. Check if you're on operator dashboard
2. Add `NEXT_PUBLIC_SHOW_DEBUG_TOOLS=true` to Render
3. Deploy again with cache clear



After Confirmation

Once everything is working and you've confirmed events are appearing:

1. Disable Debug Mode (Optional)

Edit these files to reduce log noise:

sentry.client.config.ts:

```
// Line 38 - Change from:  
debug: true,  
// To:  
debug: ENVIRONMENT === 'development',
```

sentry.server.config.ts:

```
// Line 31 - Change from:  
debug: true,  
// To:  
debug: ENVIRONMENT === 'development',
```

sentry.edge.config.ts:

```
// Line 32 - Change from:  
debug: true,  
// To:  
debug: ENVIRONMENT === 'development',
```

Commit and deploy:

```
git add sentry.*.config.ts  
git commit -m "fix: Disable Sentry debug mode in production"  
git push origin main
```

2. Keep or Remove Debug Tools

Option A: Keep Everything (Recommended)



- Test endpoints are useful for monitoring
- Debug button helps troubleshoot issues
- Just remove `NEXT_PUBLIC_SHOW_DEBUG_TOOLS` to hide button

Option B: Clean Up (If you prefer)

- Remove debug button from layout
- Keep test endpoints for monitoring
- Remove test page if not needed

Quick Reference

Render URLs

Dashboard: <https://dashboard.render.com>
 Logs: Your Service  Logs
 Environment: Your Service  Environment

Test URLs (Replace with your domain)

Server Test: <https://carelinkai.onrender.com/api/test-sentry-server>
 Edge Test: <https://carelinkai.onrender.com/api/test-sentry-edge>
 Client Test: <https://carelinkai.onrender.com/test-sentry-client>

Key Environment Variables

```
NEXT_PUBLIC_SENTRY_DSN=<your-dsn> # CRITICAL for client
SENTRY_DSN=<your-dsn>           # Server/Edge (optional)
NEXT_PUBLIC_SHOW_DEBUG_TOOLS=true # Show debug button (optional)
```

Git Commands

```
# Push changes
git push origin main

# Check commit
git log --oneline -1







# Check files
git diff HEAD~1
```

Documentation


- **Detailed Guide:** `SENTRY_DEBUG_GUIDE.md`
- **Quick Summary:** `SENTRY_DEBUG_SUMMARY.md`
- **This Checklist:** `SENTRY_DEPLOYMENT_CHECKLIST.md`

Expected Final State

Render Logs

```
[Sentry Debug] ===== CLIENT CONFIG =====  
[Sentry Debug] NEXT_PUBLIC_SENTRY_DSN exists: true   
[Sentry]  Client-side initialization successful  
  
[Sentry Debug] ===== SERVER CONFIG =====  
[Sentry Debug] SENTRY_DSN exists: true   
[Sentry]  Server-side initialization successful  
  
[Sentry Debug] ===== EDGE CONFIG =====  
[Sentry Debug] SENTRY_DSN exists: true   
[Sentry]  Edge initialization successful
```

Sentry Dashboard

-  Test events appearing
- Real production errors being tracked
- Full context and breadcrumbs
- Stack traces available


Application

- No visible changes to users
- Debug tools available for admins
- Error tracking working silently

Current Status:  All changes committed, ready to push and deploy

Next Action: Follow steps 1-11 above

Estimated Time: 30 minutes total

Good luck! 

If you encounter any issues, refer to `SENTRY_DEBUG_GUIDE.md` for detailed troubleshooting.