🧪 Sentry Debug Implementation - Quick Summary

Date: January 2, 2026
Status: ✅ Ready for Deployment

🎯 What Was Done

1. Enhanced All Sentry Config Files ✅

- **Files Modified:**
- `sentry.client.config.ts`
- `sentry.server.config.ts`
- `sentry.edge.config.ts`
- **Changes:**
- Added comprehensive debug logging
- Force-enabled `debug: true` (temporary)
- Added environment variable existence checks
- Added initialization status logging
- Enhanced error messages with emojis for easy log scanning

2. Created Test Endpoints ✅

- **`/api/test-sentry-server`** - Server-side testing
- **`/api/test-sentry-edge`** - Edge runtime testing
- **`/test-sentry-client`** - Full client-side test page

3. Created Debug UI Components ✅

- **Floating Debug Button** (`SentryTestButton.tsx`)
- Quick access to all Sentry tests
- Integrated into operator dashboard
- Only shown in development or when explicitly enabled

4. Comprehensive Documentation ✅

- **`SENTRY_DEBUG_GUIDE.md`** - Full debugging guide
- **`SENTRY_DEBUG_SUMMARY.md`** - This file

🔍 Key Finding: The Likely Issue

Missing `NEXT_PUBLIC_SENTRY_DSN` Environment Variable

The log message:

```
[Sentry] Not running in browser environment
```

Indicates that client-side Sentry is not initializing because `NEXT_PUBLIC_SENTRY_DSN` is not set in the Render environment variables.

Why This Matters

Variable	Available Where
`SENTRY_DSN`	Server & Edge only ✅ (currently working)
`NEXT_PUBLIC_SENTRY_DSN`	Browser, Server, & Edge ❌ (missing)

Client-side Sentry REQUIRES the `NEXT_PUBLIC_` prefix to work in the browser.

🚀 Deployment Checklist

Step 1: Add Environment Variable to Render

```
# Go to Render Dashboard → Your Service → Environment
# Add this variable:

NEXT_PUBLIC_SENTRY_DSN=<your-sentry-dsn>
```

Get DSN from: Sentry → Settings → Projects → [Your Project] → Client Keys (DSN)

Step 2: Commit and Push Changes

```
cd /home/ubuntu/carelinkai-project

git add .
git commit -m "debug: Add comprehensive Sentry debugging

- Enhanced all Sentry config files with debug logging
- Created test endpoints and UI components
- Added comprehensive documentation
- Focus: Debug client-side initialization issue"

git push origin main
```

Step 3: Deploy with Cache Clear

⚠️ **CRITICAL:** Must clear build cache!

Render → Manual Deploy → ✅ Clear build cache → Deploy

Why? Next.js bundles `NEXT_PUBLIC_` variables at BUILD time, not runtime.

Step 4: Monitor Logs

Watch for:

```
[Sentry Debug] NEXT_PUBLIC_SENTRY_DSN exists: true  ← Should be TRUE!
[Sentry] ✅ Client-side initialization successful
```

Step 5: Test All Endpoints

```
# Server test
curl https://your-app.onrender.com/api/test-sentry-server

# Edge test
curl https://your-app.onrender.com/api/test-sentry-edge

# Client test (in browser)
https://your-app.onrender.com/test-sentry-client
```

Step 6: Verify in Sentry Dashboard

Within 1-2 minutes, you should see test events with 🧪 emoji.

📊 What the Logs Will Tell You

Good Logs (Working)

```
[Sentry Debug] ==================== CLIENT CONFIG ====================
[Sentry Debug] Browser Environment: true
[Sentry Debug] NEXT_PUBLIC_SENTRY_DSN exists: true ✅
[Sentry Debug] NEXT_PUBLIC_SENTRY_DSN length: 89
[Sentry] ✅ Client-side initialization successful
[Sentry Debug] Sentry.isInitialized(): true
```

Bad Logs (Current Issue)

```
[Sentry Debug] ==================== CLIENT CONFIG ====================
[Sentry Debug] Browser Environment: false
[Sentry Debug] NEXT_PUBLIC_SENTRY_DSN exists: false ❌
[Sentry Debug] NEXT_PUBLIC_SENTRY_DSN length: 0
[Sentry] ❌ CRITICAL: NEXT_PUBLIC_SENTRY_DSN is not set
```

🧪 Testing Tools Created

Tool	URL/Location	Purpose
Server Test	`/api/test-sentry-server`	Test server-side Sentry
Edge Test	`/api/test-sentry-edge`	Test edge runtime Sentry
Client Test Page	`/test-sentry-client`	Full UI for client testing
Debug Button	Operator dashboard (bottom-right)	Quick access to tests

---

📁 Files Created/Modified

Created Files

- ✅ `src/app/api/test-sentry-server/route.ts`
- ✅ `src/app/api/test-sentry-edge/route.ts`
- ✅ `src/app/test-sentry-client/page.tsx`
- ✅ `src/components/debug/SentryTestButton.tsx`
- ✅ `SENTRY_DEBUG_GUIDE.md`
- ✅ `SENTRY_DEBUG_SUMMARY.md`

Modified Files

- ✅ `sentry.client.config.ts` - Enhanced debug logging
- ✅ `sentry.server.config.ts` - Enhanced debug logging
- ✅ `sentry.edge.config.ts` - Enhanced debug logging
- ✅ `src/app/operator/layout.tsx` - Added debug button

⚙️ Required Render Environment Variables

```
# CRITICAL: This one is likely missing
NEXT_PUBLIC_SENTRY_DSN=https://your-key@sentry.io/your-project

# OPTIONAL: This one probably already works (server/edge show success)
SENTRY_DSN=https://your-key@sentry.io/your-project

# Best practice: Set BOTH
```

🎯 Expected Outcome

After deployment with the environment variable set:

In Logs

- ✅ All three configs show successful initialization
- ✅ `NEXT_PUBLIC_SENTRY_DSN exists: true`
- ✅ Test endpoints return success

In Sentry Dashboard

- ✅ Test events appear with 🧪 emoji
- ✅ Real production errors start appearing
- ✅ Full context and breadcrumbs attached

🔄 After Confirmation

Once everything is working:

1. **Disable Debug Mode:**
   ```
   typescript
      // In all three config files, change:
      debug: true
      // Back to:
      debug: ENVIRONMENT === 'development'
   ```

2. **Optional: Hide Debug Button:**
   ```
   bash
      # Remove from Render env vars:
      NEXT_PUBLIC_SHOW_DEBUG_TOOLS
   ```

3. **Keep Test Endpoints:**
   - Useful for ongoing monitoring
   - Can restrict to admins if needed

📞 Quick Reference

Render Dashboard

```
Your Service → Environment → Add Variable
Name: NEXT_PUBLIC_SENTRY_DSN
Value: <your-sentry-dsn>
Save → Manual Deploy → Clear build cache ✅
```

Test Commands

```
# Server
curl https://your-app.onrender.com/api/test-sentry-server

# Edge
curl https://your-app.onrender.com/api/test-sentry-edge

# Client (browser)
https://your-app.onrender.com/test-sentry-client
```

Git Commands

```
git add .
git commit -m "debug: Add Sentry debugging"
git push origin main
```

📝 Important Notes

1. **Must Clear Build Cache:** Restart alone won’t work for `NEXT_PUBLIC_` variables
2. **Debug Mode is Temporary:** Produces verbose logs, disable after confirming
3. **Client-Side is the Focus:** Server/Edge already work, client needs fixing
4. **Test Endpoints are Safe:** No sensitive data exposed

✅ Success Criteria

- [ ] `NEXT_PUBLIC_SENTRY_DSN` set in Render
- [ ] Application rebuilt with cache clear
- [ ] All three configs show successful init in logs
- [ ] Test endpoints return success
- [ ] Test events appear in Sentry dashboard
- [ ] Real errors start being captured

For detailed information, see: `SENTRY_DEBUG_GUIDE.md`

Status: ✅ Ready for deployment
Next Step: Add environment variable to Render and deploy