Emergency Fix Deployment Guide

Sports Bar TV Controller - Critical Error Fixes

Date: October 21, 2025
Server: 24.123.87.42:3001
Status: CRITICAL FIXES READY FOR DEPLOYMENT

🚨 Issues Fixed

1. Missing API Endpoint (404 Error)

- **Problem:** `/api/matrix/video-input-selection` endpoint was returning 404
- **Fix:** The endpoint exists in the codebase and handles video input selection for matrix outputs
- **Impact:** Matrix video input routing will now work correctly

2. React Rendering Error (#31)

- **Problem:** "Objects are not valid as a React child" error in the browser console
- **Fix:** All database operations now use `sanitizeData()` to properly convert objects/dates to strings
- **Impact:** No more React rendering crashes when displaying Atlas configuration

3. Database Binding Errors

- **Problem:** "SQLite3 can only bind numbers, strings, bigints, buffers, and null" errors
- **Fix:** Added comprehensive data sanitization in `db-helpers.ts`
- **Impact:** All database operations work correctly without type errors

4. Atlas Hardware Query Errors

- **Problem:** "port is not defined" errors during Atlas hardware queries
- **Fix:** Improved error handling and default port values
- **Impact:** More reliable Atlas processor connectivity

📋 Deployment Options

Option 1: Automated Script (Recommended)

For Windows (PowerShell):

```
# Run this in PowerShell on the remote server
cd C:\Sports-Bar-TV-Controller  # Or your app directory
.\DEPLOY_REMOTE_FIX.ps1
```

For Linux (Bash):

```
# Run this in a terminal on the remote server
cd ~/Sports-Bar-TV-Controller  # Or your app directory
chmod +x DEPLOY_REMOTE_FIX.sh
./DEPLOY_REMOTE_FIX.sh
```

Option 2: Manual Deployment

Step 1: Connect to Remote Server

- **RDP (Windows):** Connect to 24.123.87.42:3389
- Username: Administrator
- Password: Thebar2024!
- **SSH (Linux/Mac/Windows):**
  `bash`
  `ssh -p 2222 Administrator@24.123.87.42`

Step 2: Navigate to Application Directory

```
cd Sports-Bar-TV-Controller
# Or find it with: find ~/ -name "Sports-Bar-TV-Controller" -type d
```

Step 3: Pull Latest Code

```
# Stash any local changes
git stash

# Pull from main branch
git fetch origin
git checkout main
git pull origin main
```

Step 4: Install Dependencies (if package.json changed)

```
npm install
```

Step 5: Build the Application

```
npm run build
```

Step 6: Restart the Application

If using PM2:

```
pm2 list                    # See running apps
pm2 restart sports-bar-tv   # Restart your app
pm2 save                    # Save the configuration
```

If using systemd:

```
sudo systemctl restart sports-bar
```

Manual restart:
1. Stop the current process (Ctrl+C if running in terminal, or kill the process)
2. Start again: `npm run start`

Step 7: Verify the Fix

1. Open http://24.123.87.42:3001/audio-control in your browser
2. Open the browser console (F12)
3. Verify no React errors appear
4. Test the Atlas processor connection
5. Try selecting video inputs for matrix outputs

🔍 Verification Checklist

After deployment, verify the following:

- [ ] Application loads without “Something went wrong!” error
- [ ] No “Minified React error #31” in browser console
- [ ] “Real Atlas configuration loaded from hardware” appears in console (success message)
- [ ] `/api/matrix/video-input-selection` endpoint works (no 404)
- [ ] Audio Zone Control displays zones correctly
- [ ] Matrix input selection works
- [ ] No “SQLite3 can only bind...” errors in server logs

📊 Technical Details

Files Modified/Fixed

1. **`src/lib/db-helpers.ts`**
   - Added `sanitizeData()` function to convert all data types to SQLite-compatible formats
   - Applied to all database operations (create, update, createMany)

2. **`src/app/api/matrix/video-input-selection/route.ts`**
   - Complete implementation of video input selection API
   - Handles routing Wolfpack matrix inputs to outputs

3. **`src/app/api/audio-processor/[id]/zones-status/route.ts`**
   - Fixed port parameter handling
   - Added default values for tcpPort and httpPort

4. **`src/lib/atlas-hardware-query.ts`**
   - Improved error handling
   - Better connection management

Code Changes Summary

sanitizeData() function in db-helpers.ts:

```
function sanitizeData(data: any): any {
  const sanitized: any = {}
  for (const [key, value] of Object.entries(data)) {
    if (value === undefined) {
      continue
    } else if (value instanceof Date) {
      sanitized[key] = value.toISOString()  // Convert Date to string
    } else if (typeof value === 'boolean') {
      sanitized[key] = value ? 1 : 0  // Convert boolean to number
    } else if (typeof value === 'object') {
      sanitized[key] = JSON.stringify(value)  // Convert objects to JSON
    } // ... other type conversions
  }
  return sanitized
}
```

🆘 Troubleshooting

Application Won't Start

1. Check if port 3001 is already in use: `netstat -ano | findstr :3001` (Windows) or `lsof -i :3001` (Linux)
2. Check Node.js is installed: `node --version`
3. Check npm dependencies: `npm install --force`

Still Seeing Errors

1. Clear browser cache and hard reload (Ctrl+Shift+R)
2. Check server logs: `pm2 logs` or check console output
3. Verify Git pulled latest code: `git log -1` (should show recent commits)

Database Errors Persist

1. Backup the database: `cp sports-bar.db sports-bar.db.backup`
2. Run migrations: `npm run db:push` (if available)
3. Check database permissions

Can't Connect to Server

1. Verify server is running: `curl http://24.123.87.42:3001`
2. Check firewall settings
3. Verify network connectivity

📞 Support

If you encounter issues after deployment:

1. **Check Browser Console:**
   - Press F12 to open Developer Tools
   - Go to Console tab
   - Screenshot any error messages

2. **Check Server Logs:**
   - PM2: `pm2 logs sports-bar-tv`
   - Or check application console output

3. **Gather Information:**
   - Git commit hash: `git rev-parse HEAD`
   - Node version: `node --version`
   - NPM version: `npm --version`
   - Application status: `pm2 list` or `systemctl status sports-bar`

🎯 Expected Outcome

After successful deployment:

✅ Application loads cleanly without crashes
✅ Atlas processor configuration displays correctly
✅ Video input selection API works
✅ No React rendering errors
✅ No SQLite binding errors
✅ Improved error handling and logging

📝 Deployment Log Template

```
Deployment Date: _____
Deployed By: _____
Git Commit: _____

Pre-Deployment Status:
[ ] Application running
[ ] Errors present
[ ] Database accessible

Deployment Steps:
[ ] Code pulled from GitHub
[ ] Dependencies installed
[ ] Application built
[ ] Application restarted

Post-Deployment Verification:
[ ] Application loads
[ ] No React errors
[ ] API endpoints working
[ ] Atlas connectivity working

Notes:
_____
_____
```

IMPORTANT: After deployment, monitor the application for 10-15 minutes to ensure stability.