

Manual Deployment Steps

Prerequisites

- SSH access to the remote server (24.123.187.42:224)
- `sshpass` installed on your local machine (or use your SSH client of choice)

Quick Deployment

Option 1: Using the Automated Script (Recommended)

```
./deploy-to-remote.sh
```

Option 2: Manual Step-by-Step Deployment

Step 1: SSH into the Remote Server

```
ssh -p 224 ubuntu@24.123.187.42  
# Password: 6809233DjD$$$
```

Step 2: Navigate to Application Directory

```
cd ~/Sports-Bar-TV-Controller  
# Or check other common locations:  
# cd /var/www/Sports-Bar-TV-Controller  
# cd /opt/Sports-Bar-TV-Controller
```

Step 3: Backup Current State (Optional but Recommended)

```
# Stash any local changes  
git stash save "Pre-deployment backup $(date)"  
  
# Or create a backup branch  
git branch backup-$(date +%Y%m%d-%H%M%S)
```

Step 4: Pull Latest Changes from GitHub

```
# Fetch latest changes  
git fetch origin  
  
# Switch to main branch  
git checkout main  
  
# Pull latest code  
git pull origin main
```

Step 5: Install Dependencies (if package.json changed)

```
npm install
```

Step 6: Build the Application

```
npm run build
```

Step 7: Restart the Application

If using PM2:

```
# List running processes
pm2 list

# Restart the application
pm2 restart sports-bar-tv
# or
pm2 restart all

# Save PM2 configuration
pm2 save
```

If using systemd:

```
sudo systemctl restart sports-bar-tv
sudo systemctl status sports-bar-tv
```

If running manually:

```
# Stop the current process (Ctrl+C if in foreground)
# Then start again:
npm start
# or for production:
npm run start
```

Step 8: Verify Deployment

```
# Check if the application is running on port 3000
netstat -tuln | grep 3000

# Test the application endpoint
curl http://localhost:3000/

# Test Atlas processor API
curl http://localhost:3000/api/audio-processor/atlas-001
```

Step 9: Test from Browser

Open your browser and navigate to:

- Main application: <http://24.123.187.42:3000>
- Audio Control: <http://24.123.187.42:3000/audio-control>

Verify:

1. ☒ Application loads without errors
2. ☒ Atlas processor shows as “Authenticated” or “Connected”
3. ☒ No “Processor ID is required” error
4. ☒ Audio zones are visible and configurable
5. ☒ Input gain controls work without 500 errors

Changes Deployed

Atlas Processor Fixes

- ☒ Fixed connection protocol to Atlas processor
- ☒ Updated API endpoints to use proper async/await patterns
- ☒ Fixed Next.js 15 compatibility issues with dynamic route parameters
- ☒ Removed mock data and implemented real Atlas communication

API Endpoint Updates

- ☒ `/api/audio-processor/[id]/input-gain` - Fixed to properly handle processor ID
- ☒ `/api/audio-processor/[id]/ai-gain-control` - Updated for async params
- ☒ `/api/audio-processor/[id]/ai-monitoring` - Updated for async params
- ☒ `/api/audio-processor/[id]/adjustment-history` - Updated for async params

Frontend Improvements

- ☒ Fixed undefined property errors in rendering
- ☒ Improved error handling and display
- ☒ Better connection status indicators

Troubleshooting

Application Won't Start

```
# Check logs
pm2 logs
# or
journalctl -u sports-bar-tv -n 50 -f

# Check for port conflicts
lsof -i :3000

# Verify Node.js version
node --version # Should be 18.x or higher
```

Atlas Processor Connection Issues

1. Verify Atlas processor is accessible:

```
bash
curl -v http://192.168.5.101:5321
```

2. Check network configuration:

- Ensure the server can reach 192.168.5.101:5321
- Check firewall rules: `sudo ufw status`

3. Test from application server:

```
bash
nc -zv 192.168.5.101 5321
```

Build Failures

```
# Clear npm cache
npm cache clean --force

# Remove node_modules and reinstall
rm -rf node_modules
npm install

# Clear Next.js cache
rm -rf .next

# Rebuild
npm run build
```

Database Issues

```
# Check if Prisma needs migration
npx prisma migrate status

# Apply migrations if needed
npx prisma migrate deploy

# Regenerate Prisma client
npx prisma generate
```

Rollback Procedure

If the deployment causes issues:

```
# Find your backup or previous commit
git log --oneline -10

# Rollback to previous commit
git reset --hard <commit-hash>

# Or restore from stash
git stash list
git stash apply stash@{0}

# Rebuild and restart
npm run build
pm2 restart all
```

Support

If you encounter issues:

1. Check the console logs in the browser (F12)
2. Check server logs: `pm2 logs` or application logs

3. Verify Atlas processor is running and accessible
4. Check network connectivity between server and Atlas processor

Post-Deployment Checklist

- ☐ Application loads at <http://24.123.187.42:3000>
 - ☐ Audio Control page accessible at </audio-control>
 - ☐ Atlas processor shows as authenticated
 - ☐ No console errors in browser developer tools
 - ☐ Audio zones load correctly
 - ☐ Input gain controls respond without 500 errors
 - ☐ AI monitoring displays data (if applicable)
 - ☐ Zone controls work as expected
-

Deployment Date: October 19, 2025

Repository: <https://github.com/dfultonthebar/Sports-Bar-TV-Controller>

Branch: main

Last Commit: Merge fix/input-gain-500-error - Atlas processor connection fixes