



Channel Preset Deployment Fix Guide

Issues Fixed







1.  **Duplicate Import:** Already fixed in commit `f23e0af` - no action needed
2.  **Migration Order:** Fixed by renaming migrations with proper timestamps

Quick Deployment (Recommended)

Automated Script (Easiest)

```
cd /home/ubuntu/Sports-Bar-TV-Controller
git fetch origin
git checkout fix/migration-order-and-import
git pull origin fix/migration-order-and-import
chmod +x deploy-fix.sh
./deploy-fix.sh
```

The script will:

-  Create database backup
-  Pull latest fixes
-  Resolve migration conflicts
-  Apply migrations in correct order
-  Build application
-  Restart services

Manual Deployment Options

Option 1: Fresh Migration (No Production Data)

Use this if you can reset the database without losing important data.

```
cd /home/ubuntu/Sports-Bar-TV-Controller
git fetch origin
git checkout fix/migration-order-and-import
git pull origin fix/migration-order-and-import

# Reset and apply migrations
npx prisma migrate reset --force
npx prisma migrate deploy

# Build and restart
npm run build
pm2 restart all # or: sudo systemctl restart sports-bar-tv
```

Option 2: Preserve Existing Data

Use this if you have production data to preserve.

```
cd /home/ubuntu/Sports-Bar-TV-Controller

# Backup database first!
cp prisma/data/sports_bar.db prisma/data/sports_bar.db.backup

# Pull fixes
git fetch origin
git checkout fix/migration-order-and-import
git pull origin fix/migration-order-and-import

# Mark old migration as applied (skip it)
npx prisma migrate resolve --applied 20250103_add_usage_tracking

# Apply new migrations
npx prisma migrate deploy

# Build and restart
npm run build
pm2 restart all # or: sudo systemctl restart sports-bar-tv
```

Verification Steps

After deployment, verify everything works:

1. Check Application Status



```
# If using PM2:
pm2 status
pm2 logs --lines 50

# If using systemd:
systemctl status sports-bar-tv
journalctl -u sports-bar-tv -n 50 -f
```

2. Check Database Migrations

```
cd /home/ubuntu/Sports-Bar-TV-Controller
npx prisma migrate status
```

Expected output should show:

-  20250103000001_channel_presets - Applied
-  20250103000002_add_usage_tracking - Applied

3. Test in Web Interface

1. Open your Sports Bar TV Controller web interface
 2. Navigate to the Bartender Remote
 3. Verify channel preset buttons are visible
 4. Test clicking a preset button
 5. Check that channels switch correctly
-

Troubleshooting

Build Errors

If you see TypeScript or build errors:

```
cd /home/ubuntu/Sports-Bar-TV-Controller
rm -rf node_modules package-lock.json
npm install
npm run build
```

Migration Errors

If migrations fail:

```
# Check migration status
npx prisma migrate status

# View detailed error
npx prisma migrate deploy --help

# Last resort: reset (LOSES DATA!)
npx prisma migrate reset --force
```

Application Won't Start

```
# Check logs
pm2 logs --lines 100

# Or for systemd:
journalctl -u sports-bar-tv -n 100 -f

# Try manual start to see errors
npm run dev
```

What Changed

Migration Files Renamed

- `20250103_channel_presets` → `20250103000001_channel_presets`
- Creates the ChannelPreset table first
- `20250103_add_usage_tracking` → `20250103000002_add_usage_tracking`
- Adds usage tracking columns after table exists

Why This Fixes the Issue

Prisma applies migrations in alphabetical order. The old names caused:

- `20250103_add_usage_tracking` to run first (alphabetically before `channel_presets`)
- Error: "no such table: ChannelPreset"

The new timestamps ensure correct order:

1. `20250103000001_channel_presets` creates table
2. `20250103000002_add_usage_tracking` adds columns

Support

- **GitHub PR:** <https://github.com/dfultonthebar/Sports-Bar-TV-Controller/pull/55>
- **Issues:** <https://github.com/dfultonthebar/Sports-Bar-TV-Controller/issues>

Database Backup Location

Backups are automatically created at:

```
/home/ubuntu/Sports-Bar-TV-Controller/backups/YYYYMMDD_HHMMSS/
```