

Phase 2: Save Configuration API Fix

Date: October 10, 2025

Branch: fix-save-config-api

Pull Request: #181

Status:  COMPLETE

Issues Fixed

1. Non-existent Field References →

Problem:

```
// Old code tried to save fields that don't exist in database
selectedVideoInput: output.selectedVideoInput || null,
videoInputLabel: output.videoInputLabel || null
```

Error:

The column `main.MatrixOutput.selectedVideoInput` does not exist in the current database.

Solution:

- Removed references to non-existent fields
- Used raw SQL for outputs to include database-only fields (`dailyTurnOn` , `dailyTurnOff` , `isMatrixOutput`)
- Added explicit select in GET to exclude non-existent fields

2. Improper UUID Generation →

Problem:

```
// Old code used empty string as fallback
where: { id: config.id || '' }
```

Impact: Invalid IDs causing upsert failures

Solution:

```
// Generate proper UUIDs
import { randomUUID } from 'crypto'
const configId = config.id || randomUUID()
```

3. Missing Transaction Wrapper →

Problem:

- Delete and create operations not atomic
- Database could be left in inconsistent state on errors

Solution:

```
const result = await prisma.$transaction(async (tx) => {
  // All operations wrapped in transaction
  await tx.matrixConfiguration.upsert(...)
  await tx.matrixInput.deleteMany(...)
  await tx.matrixOutput.deleteMany(...)
  await tx.matrixInput.createMany(...)
  // Raw SQL for outputs
  return savedConfig
})
```

4. Duplicate PrismaClient ❌ → ✅

Problem:

```
// matrix-config/route.ts was creating new instance
import { PrismaClient } from '@prisma/client'
const prisma = new PrismaClient()
```

Impact: Connection pool issues and race conditions

Solution:

```
// Use singleton
import prisma from '@lib/prisma'
```

5. Wrong Relation Names ❌ → ✅

Problem:

```
// GET endpoint used wrong names
include: {
  MatrixInput: { ... },
  MatrixOutput: { ... }
}
```

Error:

```
Unknown field `MatrixInput` for include statement
```

Solution:

```
// Use correct Prisma relation names
include: {
  inputs: { ... },
  outputs: { ... }
}
```

6. Missing Validation ❌ → ✅

Problem: No validation of required fields

Solution:

```
if (!config.name || !config.ipAddress) {
  return NextResponse.json({
    error: 'Missing required fields: name and ipAddress are required'
  }, { status: 400 })
}
```

7. Multiple Active Configs ❌ → ✅

Problem: Could have multiple active configurations

Solution:

```
// Auto-deactivate other configs when saving active one
if (config.isActive !== false) {
  await tx.matrixConfiguration.updateMany({
    where: { id: { not: configId } },
    data: { isActive: false }
  })
}
```

8. Poor Error Handling ❌ → ✅

Problem: Generic error messages

Solution:

```
// Detailed error messages and console logging
console.log(`Configuration saved successfully: ${result.name} (${result.id})`)
console.log(`- Inputs saved: ${inputs?.length || 0}`)
console.log(`- Outputs saved: ${outputs?.length || 0}`)

return NextResponse.json({
  error: 'Failed to save configuration',
  details: errorMessage
}, { status: 500 })
```

Files Modified

src/app/api/matrix/config/route.ts

Changes:

- Complete rewrite of POST handler
- Added transaction wrapper
- Added proper UUID generation
- Fixed field references
- Added validation
- Enhanced error handling
- Fixed GET handler with explicit selects

Lines changed: 113 insertions, 73 deletions

src/app/api/matrix-config/route.ts

Changes:

- Fixed PrismaClient singleton import
- Removed unnecessary \$disconnect

Lines changed: 7 insertions, 7 deletions

Testing Results

Test 1: Save Configuration





Request:

```
{
  "config": {
    "name": "Graystone Matrix Test",
    "ipAddress": "192.168.5.100",
    "tcpPort": 23,
    "protocol": "TCP",
    "isActive": true
  },
  "inputs": [
    {"channelNumber": 1, "label": "Cable Box 1", "deviceType": "Cable Box"},
    {"channelNumber": 2, "label": "Cable Box 2", "deviceType": "Cable Box"},
    {"channelNumber": 5, "label": "Direct TV 1", "deviceType": "Direct TV"}
  ],
  "outputs": [
    {"channelNumber": 1, "label": "TV 01"},
    {"channelNumber": 2, "label": "TV 02"},
    {"channelNumber": 33, "label": "Matrix 1"}
  ]
}
```

Response:

```
{
  "success": true,
  "message": "Configuration saved successfully",
  "config": {
    "id": "9b287296-6bad-481d-a594-efb71d339918",
    "name": "Graystone Matrix Test",
    "ipAddress": "192.168.5.100",
    "tcpPort": 23,
    "udpPort": 4000,
    "protocol": "TCP",
    "isActive": true
  },
  "inputCount": 3,
  "outputCount": 3
}
```

Database verification:

- MatrixConfiguration: 1 record 
- MatrixInput: 3 records 
- MatrixOutput: 3 records 
- isActive flag: true 

Test 2: Load Configuration

Request: GET /api/matrix/config


Response:

```
{
  "configs": [...],
  "config": {
    "id": "9b287296-6bad-481d-a594-efb71d339918",
    "name": "Graystone Matrix Test",
    "ipAddress": "192.168.5.100",
    "inputs": [
      {
        "channelNumber": 1,
        "label": "Cable Box 1",
        "deviceType": "Cable Box",
        "isActive": true
      },
      ...
    ],
    "outputs": [
      {
        "channelNumber": 1,
        "label": "TV 01",
        "resolution": "1080p",
        "isActive": true
      },
      ...
    ]
  }
}
```

Test 3: PM2 Restart Persistence

Steps:

1. Save configuration
2. Restart PM2: `pm2 restart sports-bar-tv-controller`
3. Load configuration

Result: Configuration persisted 

Test 4: Multiple Saves

Steps:

1. Save configuration A
2. Save configuration B with `isActive: true`
3. Check database

Result: Only configuration B is active 

Database State After Fix

MatrixConfiguration records: 1

MatrixInput records: 3

MatrixOutput records: 3

Active configurations: 1

Sample data:

Configuration: Graystone Matrix Test | 192.168.5.100 | TCP | Active

Inputs:

- 1: Cable Box 1 (Cable Box)
- 2: Cable Box 2 (Cable Box)
- 5: Direct TV 1 (Direct TV)

Outputs:

- 1: TV 01
- 2: TV 02
- 33: Matrix 1

Deployment Instructions

On Server (24.123.87.42)

```
cd ~/Sports-Bar-TV-Controller

# Pull the fix branch
git fetch origin fix-save-config-api
git checkout fix-save-config-api

# Install dependencies
npm ci

# Build application
npm run build

# Restart PM2
pm2 restart sports-bar-tv-controller

# Verify
curl http://localhost:3001/api/matrix/config
```

Verification Steps

1. Test Save:

```
bash
curl -X POST http://localhost:3001/api/matrix/config \
-H "Content-Type: application/json" \
-d '{"config":{"name":"Test","ipAddress":"192.168.1.100"},"inputs":[],"outputs":[]}'
```

2. Check Database:

```
bash
sqlite3 ./prisma/data/sports_bar.db "SELECT * FROM MatrixConfiguration;"
```

3. Test Load:

```
bash
curl http://localhost:3001/api/matrix/config
```

4. Test Persistence:

```
bash
pm2 restart sports-bar-tv-controller
```

```
sleep 3
curl http://localhost:3001/api/matrix/config
```

Known Limitations

Schema Mismatch

The Prisma schema and database schema are out of sync:

Prisma has but DB doesn't:

- selectedVideoInput
- videoInputLabel

DB has but Prisma doesn't:

- dailyTurnOn
- dailyTurnOff
- isMatrixOutput

Workaround:

- Use raw SQL for outputs in POST
- Use explicit select in GET to exclude non-existent fields

Proper fix: Sync Prisma schema with database or run migrations

Success Criteria

- [x] Save Configuration button works
- [x] Configuration persists in database with isActive: true
- [x] Configuration survives PM2 restart
- [x] Tests can find the active configuration
- [x] GET endpoint returns configuration
- [x] Inputs saved correctly with proper IDs
- [x] Outputs saved correctly with proper IDs
- [x] Error messages are clear and helpful
- [x] Only one active configuration at a time
- [x] Transaction ensures atomicity

Next Steps (Phase 3)

1. Enter correct Graystone Matrix configuration:
 - Name: "Graystone Matrix"
 - IP: 192.168.5.100
 - Protocol: TCP, Port: 23
 - 18 active inputs (Cable Box, Direct TV, Fire TV, etc.)
 - 29 active outputs (TVs 1-25, Matrix 1-4)
2. Verify configuration in UI
3. Test matrix switching functionality
4. Create regular backup schedule

5. Consider Prisma schema migration to fix mismatch

Related Issues

- Fixes “No active matrix configuration found” error
- Fixes Bartender Remote showing “Matrix: disconnected”
- Fixes configuration loss after updates
- Fixes Save Configuration button not working