System Update Enhancements

Overview

The update system has been enhanced to automatically check for and install required dependencies, including Local AI (Ollama) and color scheme standardization tools.

What Was Updated

1. Enhanced Update Script (`update_from_github.sh`)

New Features

Automatic Ollama Installation

- **Checks** if Ollama (Local AI) is installed
- **Installs** automatically if missing
- **Starts** the Ollama service if not running
- **Pulls** recommended AI models (llama3.2:3b or llama3.2)
- **Uses** the existing `install-local-ai.sh` script if available

Automatic Color Scheme Analysis

- **Runs** AI-powered style analyzer in background after updates
- **Non-blocking** - doesn’t delay app startup
- **Timeout protection** - limits analysis to 2 minutes max
- **Logs results** to `ai-style-analysis.log`
- **Generates reports** in `ai-style-reports/` directory

Enhanced Status Reporting

- Shows what was updated/verified
- Provides links to style analysis tools
- Includes helpful next steps

2. AI Configuration Page (`src/app/ai-config/page.tsx`)

New Section: AI Style Standardization

Added a comprehensive section that includes:

Features Display

- ✓ Component scanning for styling inconsistencies
- ✓ Comparison against standardized dark theme
- ✓ Detailed report generation
- ✓ Automatic fix application with backups

Quick Commands

Shows the exact commands users can run:
- Interactive menu: `./scripts/run-style-analysis.sh`
- Direct analysis: `node scripts/ai-style-analyzer.js`
- Apply fixes: `node scripts/ai-style-fixer.js`

Auto-Run Notification

Informs users that the analyzer runs automatically during updates

Quick Actions

- Link to manage API Keys
- Link to test team search
- Link to view style guide (COLOR_SCHEME_STANDARD.md)

Update Workflow

When You Run `update_from_github.sh`:

```
🔄 Updating Sports Bar AI Assistant from GitHub
📊 Check git status
⏹️  Stop running processes
⬇️  Pull latest changes from GitHub
📦 Install/update dependencies
📺 Check/install libCEC
🤖 Check/install Ollama (Local AI)
   ├─ Install if missing
   ├─ Start service if not running
   └─ Pull AI models
🗄️  Update database
🏗️  Build application
🎨 Run AI Color Scheme Analysis (background)
   ├─ Analyze components
   └─ Generate reports
🚀 Restart application
✅ Verify everything works
```

Benefits

For Users

- **One-command setup** - Everything installs automatically
- **Always up-to-date** - AI and dependencies checked on every update
- **Non-intrusive** - Background analysis doesn’t block startup
- **Clear feedback** - Know exactly what’s happening

For Developers

- **Consistent environment** - All systems have same setup
- **Automated quality checks** - Style analysis on every update
- **Easy troubleshooting** - Detailed logs and reports
- **Self-documenting** - Update script shows what it does

Usage

Running Updates

```
cd /home/ubuntu/Sports-Bar-TV-Controller
./update_from_github.sh
```

Viewing Style Analysis Results

```
# Check the log
cat ai-style-analysis.log

# View reports
ls -lht ai-style-reports/

# Run interactive tools
./scripts/run-style-analysis.sh
```

Accessing via Web UI

1. Navigate to **AI Configuration** page
2. Scroll to **AI Style Standardization** section
3. See commands and status
4. Click **View Style Guide** to see standards

Files Modified

- `update_from_github.sh` - Enhanced with AI checks and style analysis
- `src/app/ai-config/page.tsx` - Added style standardization section

Files Referenced

- `install-local-ai.sh` - Used by update script for Ollama installation
- `scripts/ai-style-analyzer.js` - Run during updates
- `scripts/run-style-analysis.sh` - Interactive menu for users
- `COLOR_SCHEME_STANDARD.md` - Style guide linked from UI

Technical Details

Ollama Detection

```
if ! command -v ollama &> /dev/null; then
    # Install Ollama
fi
```

Service Check

```
if ! pgrep -x ollama > /dev/null; then
    # Start service
fi
```

Model Check

```
if ! ollama list | grep -q "llama3.2"; then
    # Pull model
fi
```

Background Analysis

```
timeout 120 node scripts/ai-style-analyzer.js > ai-style-analysis.log 2>&1 &
```

Safety Features

Non-Blocking Design

- Style analysis runs in background
- Timeout prevents hanging
- App starts regardless of analysis status

Error Handling

- Checks for script existence before running
- Graceful degradation if tools unavailable
- Clear error messages

Log Files

- `ai-style-analysis.log` - Analysis output
- `server.log` - Application logs
- `ai-style-reports/*.json` - Detailed reports

Future Enhancements

Potential improvements:

1. **Email notifications** - Send analysis results via email
2. **Slack integration** - Post reports to Slack channel
3. **Auto-fix option** - Apply fixes automatically if severity low
4. **Dashboard widget** - Show style health in main UI
5. **CI/CD integration** - Run on pull requests
6. **Historical tracking** - Track improvement over time

Troubleshooting

Ollama Not Installing

```
# Manual installation
curl -fsSL https://ollama.ai/install.sh | sh
ollama pull llama3.2
```

Analysis Not Running

```
# Check if Ollama is available
ollama list

# Run analysis manually
node scripts/ai-style-analyzer.js
```

Reports Not Generated

```
# Check the log
cat ai-style-analysis.log

# Verify directory exists
ls -la ai-style-reports/
```

Conclusion

The enhanced update system ensures that:
- ✅ Local AI is always available
- ✅ Style consistency is monitored
- ✅ Users have easy access to tools
- ✅ System is self-maintaining
- ✅ Updates are seamless

Run your next update to see these enhancements in action!

Created: October 1, 2025
Version: 2.0
Repository: https://github.com/dfultonthebar/Sports-Bar-TV-Controller