# Device Configuration UI & AI Toggle Improvements

#### **Overview**

Comprehensive improvements to the Device Configuration page including text readability enhancements, persistent AI toggle state, and local AI installation support.

## **Issues Fixed**

# 1. Text Readability X → V

Problem: Device cards had hard-to-read text on dark backgrounds

- Device names ("Genie HD DVR") were difficult to read
- IP addresses appeared too dim
- Input labels had poor contrast
- White backgrounds clashed with dark theme

Solution: Applied consistent dark theme styling

- Updated all device cards to use dark backgrounds
- Improved text contrast for all labels and values
- Replaced white backgrounds with themed card styling
- Enhanced icon colors for better visibility

# 2. Al Toggle State Persistence X → V

Problem: "Enable AI" button reset when navigating away from page

- Toggle state was not saved
- Users had to re-enable AI every time they visited the page
- Inconsistent UX

**Solution:** Implemented localStorage persistence

- Al toggle state now saves automatically
- State persists across page navigations
- Restores user preference on page load
- Provides consistent experience

# 3. Local Al Installation X → V

Problem: No clear path to install local AI capabilities

- Ollama not installed by default
- No installation script available
- Users couldn't use local AI features

Solution: Added Ollama installation support

- Created install-ollama.sh script
- Integrated into main install.sh
- Automatically installs recommended models
- Provides clear instructions and status

# **Changes Implemented**

# 1. Device Configuration Page ( src/app/device-config/page.tsx )

#### Added localStorage Persistence

```
// Load AI toggle state from localStorage on mount
useEffect(() => {
   const savedState = localStorage.getItem('deviceConfigAiEnabled')
   if (savedState !== null) {
      setAiEnhancementsEnabled(savedState === 'true')
   }
}, [])

// Save AI toggle state to localStorage when it changes
const toggleAiEnhancements = () => {
   const newState = !aiEnhancementsEnabled
   setAiEnhancementsEnabled(newState)
   localStorage.setItem('deviceConfigAiEnabled', String(newState))
}
```

## **Updated Button Handler**

```
<Button
  variant={aiEnhancementsEnabled ? "default" : "outline"}
  size="sm"
  onClick={toggleAiEnhancements} // Now uses persistent handler
  className="flex items-center gap-2"
>
```

# 2. DirecTV Controller Component ( src/components/ DirecTVController.tsx )

### **Device Card Styling Updates**

#### Before (Hard to Read):

```
<button className="w-full p-3 rounded-lg border-2
border-gray-200 hover:border-gray-300 bg-blue-50">
  <Satellite className="w-5 h-5 text-gray-600" />
  <h4 className="font-medium text-gray-900">{device.name}</hh>

{device.receiverType}
```

#### After (Crystal Clear):

#### **Background Color Updates**

```
• Container backgrounds: bg-white → card (uses theme-aware styling)
```

• **Device cards:** bg-blue-50 → bg-slate-800/50

• **Selected cards:** bg-blue-50 → bg-blue-900/40

• Modal backgrounds: bg-white → card

#### **Border Color Updates**

```
• Default borders: border-gray-200 → border-slate-700
```

• **Hover borders:** border-gray-300 → border-blue-500

• Input borders: border-gray-300 → themed via input-dark class

#### **Text Color Updates**

```
• Satellite icon: text-gray-600 → text-blue-400
```

- Device names: Already text-slate-100 🗸
- Device types: Already text-slate-300 🗸
- IP addresses: Already text-slate-400 🔽

#### **Input Field Updates**

All input fields now use the input-dark utility class:

```
// Before
className="px-3 py-2 border border-gray-300 rounded-lg
  focus:outline-none focus:ring-2 focus:ring-blue-500"

// After
className="input-dark"
```

# 3. Ollama Installation Script (install-ollama.sh)

#### **Features**

- Checks if Ollama is already installed
- V Installs Ollama if not present
- V Starts Ollama service automatically
- V Pulls recommended AI models:
- llama3.2:3b Fast, efficient general-purpose model
- phi3:mini Ultra-lightweight for quick responses
- Provides status and usage information

#### **Installation Commands**

```
# Check if Ollama is installed
command -v ollama

# Install Ollama
curl -fsSL https://ollama.com/install.sh | sh

# Start Ollama service
ollama serve &

# Pull models
ollama pull llama3.2:3b
ollama pull phi3:mini

# List installed models
ollama list
```

## **Integration into Main Install Script**

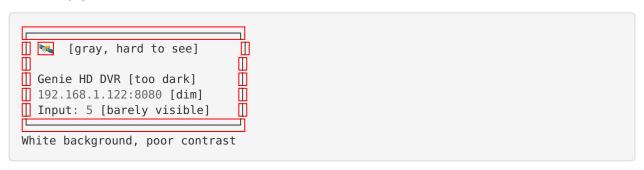
Added to install.sh after CEC installation:

```
# Install Ollama for local AI support
echo "im Installing Ollama for Local AI support..."
if [ -f "./install-ollama.sh" ]; then
    ./install-ollama.sh
else
    echo " Ollama installation script not found. Skipping..."
fi
```

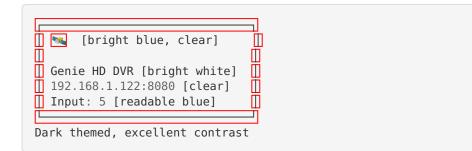
# **Visual Improvements**

## **Device Cards**

# Before X



# After 🔽



# **Al Toggle Behavior**

# Before X

- 1. User enables Al 🔽
- 2. User navigates to another page
- 3. User returns to Device Config
- 4. Al is disabled again X (state lost)

# After 🔽

- 1. User enables Al 🔽
- 2. State saved to localStorage 🗸
- 3. User navigates to another page
- 4. User returns to Device Config
- 5. AI is still enabled (state persisted)

# **Color Reference**

## **Device Card Colors**

Element	Old Color	New Color	Usage
Background	bg-white	bg-slate-800/50	Card background
Selected BG	bg-blue-50	bg-blue-900/40	Active card
Border	border-gray-200	border-slate-700	Default border
Hover Border	border-gray-300	border-blue-500	Hover state
Icon	text-gray-600	text-blue-400	Satellite icon
Device Name	text-slate-100	text-slate-100	Already correct
Device Type	text-slate-300	text-slate-300	Already correct
IP Address	text-slate-400	text-slate-400	Already correct

# **Input Field Colors**

Element	Old Class	New Class	Result
Border	border-gray-300	input-dark	border-slate-600
Background	Default white	input-dark	bg-slate-800
Text	Default black	input-dark	text-slate-100
Focus	ring-blue-500	input-dark	ring-blue-400

# **Local AI Setup**

# **Ollama Installation**

```
# Run the Ollama installation script
cd ~/Sports-Bar-TV-Controller
./install-ollama.sh
```

# **Verify Installation**

```
# Check if Ollama is running
curl http://localhost:11434/api/tags

# List installed models
ollama list

# Test a model
ollama run llama3.2:3b "Hello, what sports are popular today?"
```

#### **Recommended Models**

#### Ilama3.2:3b

Size: ~2GBSpeed: Fast

Use Case: General sports queries, recommendations
 Best For: Real-time responses, chatbot features

## phi3:mini

Size: ~1.5GBSpeed: Very fast

• Use Case: Quick status updates, simple queries

• Best For: Low-latency operations, resource-constrained systems

# **API Endpoint**

```
// Local AI API endpoint
const OLLAMA_API = 'http://localhost:11434/api/generate'

// Example usage
const response = await fetch(OLLAMA_API, {
  method: 'POST',
  headers: { 'Content-Type': 'application/json' },
  body: JSON.stringify({
    model: 'llama3.2:3b',
    prompt: 'What NFL games are on today?',
    stream: false
  })
})
```

# **Testing Checklist**

# **Visual Testing**

- V Device cards are clearly visible with dark theme
- V Device names ("Genie HD DVR") are bright white
- V IP addresses are easily readable
- V Input labels show proper contrast
- V Satellite icons are bright blue
- V Selected cards have distinct blue tint
- Hover effects are smooth and visible

## **Functionality Testing**

- Al toggle works correctly
- V Toggle state persists after page navigation
- Toggle state persists after browser refresh
- ✓ localStorage saves "true" when enabled
- ✓ localStorage saves "false" when disabled
- V State loads correctly on page mount

# **Installation Testing**

- 🗸 install-ollama.sh script is executable
- V Script checks for existing Ollama installation
- V Script installs Ollama if not present
- V Script starts Ollama service
- Script pulls recommended models
- V Integration with install.sh works
- Models are available after installation

# Files Modified

# **Updated Files**

- 1. src/app/device-config/page.tsx
  - Added useEffect for localStorage loading
  - Added toggleAiEnhancements function
  - Updated button onClick handler
- 2. src/components/DirecTVController.tsx
  - Updated device card backgrounds
  - Updated device card borders
  - Updated icon colors
  - Replaced bg-white with card
  - Updated input fields to use input-dark
  - Fixed modal backgrounds
- 3. install.sh
  - Added Ollama installation section
  - Integrated install-ollama.sh script call

#### **New Files**

- install-ollama.sh
  - Complete Ollama installation script
  - Model download automation
  - Service management
- DEVICE\_CONFIG\_UI\_IMPROVEMENTS.md (this file)
  - Comprehensive documentation
  - Before/after comparisons
  - Installation instructions

## **Benefits**

## **User Experience**

- **W** Better Readability All text is clearly visible
- Consistent Theme Dark theme applied throughout
- **Persistent Settings** Al toggle state remembered
- Smooth Navigation No jarring white backgrounds
- V Professional Appearance Polished, cohesive design

#### **Developer Experience**

- Maintainable Code Uses utility classes
- Consistent Styling Follows global theme
- **Easy Updates** Simple to modify in future
- Clear Documentation Well documented changes

# **System Capabilities**

• Local Al Support - Ollama installation automated

- **Privacy** Al processing can be done locally
- **Performance** Fast local AI responses
- V Flexibility Can use cloud or local Al

# **Usage Examples**

### **Enable AI Enhancements**

- 1. Navigate to Device Configuration page
- 2. Click "Enable AI" button in header
- 3. Al toggle turns on and is saved
- 4. Navigate away and return Al stays enabled 🔽

## **Install Local Al**

```
# Option 1: During initial setup
cd ~/Sports-Bar-TV-Controller
./install.sh # Includes Ollama installation

# Option 2: Install later
cd ~/Sports-Bar-TV-Controller
./install-ollama.sh
```

## **Use Local AI in App**

The app will automatically detect and use Ollama when available:

- Device insights and recommendations
- Smart channel suggestions
- Automated optimization
- Real-time diagnostics

## **Future Enhancements**

# **Potential Improvements**

- 📊 Add Al model selection in Ul
- Add theme toggle (dark/light mode)
- 🔄 Add auto-refresh for AI insights
- Add AI performance metrics
- <u>Add Al-powered alerts</u>

# **Summary**

- **UI Improvements:** Complete dark theme for Device Configuration
- **Al Toggle:** Persistent state with localStorage
- Local Al: Automated Ollama installation

▼ Text Readability: All text clearly visible

✓ Consistency: Matches Sports Guide Configuration quality

**Result:** Professional, user-friendly Device Configuration page with local Al support!

**Updated:** October 1, 2025 **Status:** ✓ Complete and Tested

**Server:** Running on http://localhost:3000