Deployment Instructions - Auto TV Documentation Feature

Overview

This document provides step-by-step instructions for deploying the Automatic TV Documentation feature to your Sports Bar TV Controller system.

Prerequisites

- Node.js 18+ installed
- npm or yarn package manager
- Prisma database configured and running
- Write permissions to the project directory
- Internet connectivity for downloading manuals

Installation Steps

1. Pull Latest Changes

```
cd ~/Sports-Bar-TV-Controller
git fetch origin
git checkout feat/auto-tv-docs
git pull origin feat/auto-tv-docs
```

2. Install Dependencies

```
npm install
```

This will install the new dependencies:
- `pdf-parse` - For extracting text from PDF manuals
- `cheerio` - For parsing HTML documentation
- `axios` - For HTTP requests

3. Create Manuals Directory

```
mkdir -p docs/tv-manuals
chmod 755 docs/tv-manuals
```

4. Run Database Migrations (if needed)

```
npx prisma generate
npx prisma migrate dev
```

5. Build the Application

```
npm run build
```

6. Test the Installation

```
# Test the TV documentation system
npx tsx scripts/test-tv-docs.ts
```

7. Restart the Application

```
# If using PM2
pm2 restart sportsbar-assistant

# If using systemd
sudo systemctl restart sportsbar-assistant

# If running manually
npm start
```

Verification

1. Check API Endpoints

Test that the new endpoints are working:

```
# Test documentation list endpoint
curl http://localhost:3000/api/cec/tv-documentation

# Test manual fetch endpoint (replace with actual TV model)
curl -X POST http://localhost:3000/api/cec/fetch-tv-manual \
  -H "Content-Type: application/json" \
  -d '{"manufacturer":"Samsung","model":"UN55TU8000"}'
```

2. Check UI Integration

1. Open the application in your browser: `http://localhost:3000`
2. Navigate to the CEC Discovery page
3. Run a CEC discovery scan
4. Check the TV Documentation panel (should appear on the page)
5. Verify that discovered TVs are listed
6. Try clicking “Fetch Manual” for a TV model

3. Check File System

```
# Verify manuals directory exists
ls -la docs/tv-manuals/

# Check for downloaded manuals (after running discovery)
ls -lh docs/tv-manuals/*.pdf
```

4. Check Logs

```
# Monitor logs for TV documentation activity
tail -f logs/app.log | grep "TV Docs"

# Check for any errors
tail -f logs/app.log | grep "ERROR"
```

Configuration

Customize Search Behavior

Edit `src/lib/tvDocs/searchManual.ts` to adjust search queries:

```
const queries = [
  `${manufacturer} ${model} manual PDF`,
  `${manufacturer} ${model} user guide PDF`,
  // Add more search patterns here
]
```

Adjust Q&A Generation

Edit `src/lib/tvDocs/generateQA.ts` to change Q&A generation settings:

```
// Maximum chunks to process
const maxChunks = Math.min(chunks.length, 10)

// Chunk size
const chunks = splitContentIntoChunks(content, 2000)
```

Configure File Size Limits

Edit `src/lib/tvDocs/downloadManual.ts`:

```
// Minimum file size (100KB)
const minSize = 100000

// Maximum file size (50MB)
const maxSize = 52428800
```

Usage

Automatic Mode (Recommended)

The system works automatically:

1. Run CEC discovery from the UI or API
2. When a TV is detected, documentation fetch starts automatically
3. Check the TV Documentation panel for progress
4. Q&A pairs are automatically added to the AI knowledge base

Manual Mode

To manually fetch documentation:

1. Navigate to TV Documentation panel in the UI
2. Find the TV model you want
3. Click “Fetch Manual” button
4. Wait for completion (may take 1-2 minutes)

API Usage

```
# Fetch manual for specific TV
curl -X POST http://localhost:3000/api/cec/fetch-tv-manual \
  -H "Content-Type: application/json" \
  -d '{
    "manufacturer": "Samsung",
    "model": "UN55TU8000",
    "forceRefetch": false
  }'

# Get all documentation
curl http://localhost:3000/api/cec/tv-documentation
```

Troubleshooting

Issue: Dependencies Not Installing

Solution:

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

# Remove node_modules and reinstall
rm -rf node_modules package-lock.json
npm install
```

Issue: Build Fails

Solution:

```
# Check TypeScript errors
npx tsc --noEmit

# Clear Next.js cache
rm -rf .next

# Rebuild
npm run build
```

Issue: Manuals Not Downloading

Possible Causes:
1. Network connectivity issues
2. Manual not available online
3. Search API not configured

Solution:

```
# Check network connectivity
curl -I https://www.google.com

# Check logs for detailed errors
tail -f logs/app.log | grep "TV Docs"

# Try manual fetch with different TV model
curl -X POST http://localhost:3000/api/cec/fetch-tv-manual \
  -H "Content-Type: application/json" \
  -d '{"manufacturer":"Sony","model":"XBR55X900H"}'
```

Issue: Q&A Pairs Not Generated

Possible Causes:

1. AI service not running
2. PDF extraction failed
3. Manual content too short

Solution:

```
# Check AI service status
curl http://localhost:3000/api/ai/status

# Check manual file
ls -lh docs/tv-manuals/

# Try extracting PDF manually
npx tsx -e "
  const pdf = require('pdf-parse');
  const fs = require('fs');
  const data = fs.readFileSync('docs/tv-manuals/Samsung_UN55TU8000_Manual.pdf');
  pdf(data).then(result => console.log(result.text.substring(0, 500)));
"
```

Issue: Permission Denied

Solution:

```
# Fix directory permissions
sudo chown -R $USER:$USER docs/tv-manuals
chmod 755 docs/tv-manuals
```

Monitoring

Check System Status

```
# Check disk usage
du -sh docs/tv-manuals/

# Count downloaded manuals
ls -1 docs/tv-manuals/*.pdf 2>/dev/null | wc -l

# Check database for Q&A pairs
sqlite3 prisma/dev.db "SELECT COUNT(*) FROM QAPair WHERE source LIKE '%Manual%';"
```

Monitor Performance

```
# Watch for documentation fetch activity
watch -n 5 'tail -20 logs/app.log | grep "TV Docs"'

# Monitor memory usage
ps aux | grep node

# Check API response times
curl -w "@-" -o /dev/null -s http://localhost:3000/api/cec/tv-documentation <<'EOF'
    time_namelookup:  %{time_namelookup}\n
       time_connect:  %{time_connect}\n
    time_appconnect:  %{time_appconnect}\n
      time_redirect:  %{time_redirect}\n
   time_pretransfer:  %{time_pretransfer}\n
 time_starttransfer:  %{time_starttransfer}\n
                    ----------\n
         time_total:  %{time_total}\n
EOF
```

Rollback

If you need to rollback the changes:

```
# Switch back to main branch
git checkout main

# Reinstall dependencies
npm install

# Rebuild
npm run build

# Restart application
pm2 restart sportsbar-assistant
```

Backup

Before deploying, create a backup:

```
# Backup database
cp prisma/dev.db prisma/dev.db.backup-$(date +%Y%m%d)

# Backup configuration
tar -czf config-backup-$(date +%Y%m%d).tar.gz config/ .env

# Backup existing manuals (if any)
tar -czf manuals-backup-$(date +%Y%m%d).tar.gz docs/tv-manuals/
```

Production Deployment

For production environments:

1. **Use Environment Variables:**
   `bash`
   `export NODE_ENV=production`
   `export TV_DOCS_MAX_CHUNK_SIZE=2000`
   `export TV_DOCS_MAX_CHUNKS=10`

2. **Enable Logging:**
   `bash`
   `# Configure log rotation`
   `sudo nano /etc/logrotate.d/sportsbar-assistant`

3. **Set Up Monitoring:**
   `bash`
   `# Use PM2 monitoring`
   `pm2 install pm2-logrotate`
   `pm2 set pm2-logrotate:max_size 10M`

4. **Configure Firewall:**
   `bash`
   `# Ensure outbound HTTPS is allowed for downloading manuals`
   `sudo ufw allow out 443/tcp`

Support

For issues or questions:
- Check the troubleshooting section above
- Review logs in `logs/app.log`
- Check documentation in `docs/AUTO_TV_DOCUMENTATION.md`
- Create an issue on GitHub

Next Steps

After successful deployment:

1. Run CEC discovery to detect TVs
2. Monitor the TV Documentation panel
3. Verify Q&A pairs are being generated
4. Test the AI assistant with TV-specific questions

5. Review and adjust configuration as needed

Deployment Date: October 6, 2025
Version: 1.0.0
Feature Branch: feat/auto-tv-docs