

# Manual Push Instructions

---

The fixes have been committed locally to branch: `fix/critical-qa-timeout-json-image-issues`

## To push this branch manually:

---

1. Ensure you have valid GitHub credentials configured
2. Run:

```
bash
cd /home/ubuntu/github_repos/Sports-Bar-TV-Controller
git push -u origin fix/critical-qa-timeout-json-image-issues
```

## To create a Pull Request:

---

After pushing, create a PR with:

- **Title:** Fix critical Q&A timeout, JSON parsing, and image validation issues
- **Base branch:** main
- **Compare branch:** fix/critical-qa-timeout-json-image-issues

## PR Description:

---

This PR addresses three critical issues identified in PM2 logs:

### 1. Q&A Generation Timeouts 🕒

**Problem:** Documentation files timing out after 300 seconds

**Solution:**

- Increased timeout from 300s to 600s (10 minutes)
- Implemented content chunking (3000 chars per chunk)
- Reduced max file size from 5MB to 2MB
- Added retry logic (up to 2 retries)
- Reduced concurrency from 3 to 2

### 2. Invalid JSON Responses 📝

**Problem:** AI not returning valid JSON for some files

**Solution:**

- Enhanced prompt with explicit JSON-only instructions
- Added `format: 'json'` enforcement in Ollama API
- Lowered temperature from 0.7 to 0.3
- Improved JSON parsing to strip markdown
- Added comprehensive error logging

### 3. Invalid Image Errors 🖼️

**Problem:** Layout images showing as invalid

**Solution:**

- Added `validateImage()` function using Sharp
- Validate all uploads before saving
- Validate PDF conversions before serving

- Auto-cleanup invalid files
- Clear error messages

### Files Changed:

- `src/lib/services/qa-generator.ts` : Enhanced with chunking, retries, better JSON parsing
- `src/app/api/bartender/upload-layout/route.ts` : Added image validation
- `FIXES_APPLIED.md` : Comprehensive documentation

### Testing Recommendations:

- Test with large documentation files (>1MB)
- Verify Q&As generate without timeouts
- Check JSON parsing consistency
- Upload various image formats
- Try corrupted/invalid files

### Expected Outcomes:

- ✓ No more 300s timeouts on large files
- ✓ Consistent valid JSON responses
- ✓ No more 'invalid image' errors
- ✓ Better error handling throughout