

Manual Test Checklist - Pre-Deployment Validation

Purpose: Cover functionality that's excluded from automated tests due to Vitest collection issues.

Time Required: ~30 minutes **When to Run:** Before major releases or production deployment

Test Environment Setup

- ☐ **Browser:** Chrome or Edge (latest version)
 - ☐ **Local Dev Server:** `npm run dev` running on `http://localhost:5173`
 - ☐ **Test Data:** Valid license key with test show data
 - ☐ **Roles to Test:** Judge, Steward, Admin passcodes ready
-

Section 1: Entry Management & Scoring (HIGH PRIORITY) ⚡

1.1 Score Submission - Offline & Online

Covers: `scoreSubmission.test.ts`

Test Steps:

1. Log in as Judge
2. Navigate to Class List → Select a class → Entry List
3. Select an unscored entry → Open scoresheet
4. **Online Test:**
 - ☐ Enter valid score data (times, faults, etc.)
 - ☐ Submit score
 - ☐ ☒ Verify entry list updates immediately (optimistic update)
 - ☐ ☒ Verify score persists after page refresh
5. **Offline Test:**
 - ☐ Open DevTools → Network tab → Set to "Offline"
 - ☐ Score a different entry
 - ☐ ☒ Verify UI updates immediately even offline
 - ☐ ☒ Verify no error shown to user
 - ☐ Go back online
 - ☐ ☒ Verify queued score syncs automatically
 - ☐ ☒ Check Settings → Offline Queue shows item was processed

Expected Result: Scores submit instantly to UI, sync in background, work offline.

1.2 Entry Status Management

Covers: `entryStatusManagement.test.ts`

Test Steps:

1. Log in as Steward
2. Navigate to a class Entry List
3. **Single Entry Status Change:**
 - ☐ Click an entry → Change status to "On Deck" (yellow)
 - ☐ ☒ Entry card turns yellow immediately
 - ☐ ☒ Status persists after refresh
4. **Bulk Status Change:**
 - ☐ Select 3 entries using checkboxes
 - ☐ Change status to "Running" (green)
 - ☐ ☒ All 3 entries update simultaneously
 - ☐ ☒ Changes visible to other users in real-time (test in second browser tab)

Expected Result: Status changes apply instantly, sync across devices.

1.3 Batch Entry Operations

Covers: `entryBatchOperations.test.ts`

Test Steps:

1. Log in as Admin
2. Navigate to Admin → Competition Admin
3. **Bulk Visibility Toggle:**
 - ☐ Select 5 entries
 - ☐ Click "Toggle Visibility"
 - ☐ ☒ Verify all 5 entries hidden from public view
 - ☐ Toggle back
 - ☐ ☒ Verify entries now visible
4. **Bulk Self Check-In Enable:**
 - ☐ Select 10 entries
 - ☐ Enable "Self Check-In"
 - ☐ ☒ Verify all entries now allow exhibitor self check-in
 - ☐ ☒ Check that exhibitor role can check in these entries

Expected Result: Bulk operations apply to all selected entries efficiently.

1.4 Class Completion Tracking

Covers: `classCompletionService.test.ts`

Test Steps:

1. Log in as Judge
2. Find a class with 5 entries, none scored
3. **Track Completion Progress:**
 - ☐ Score 1st entry
 - ☐ ☒ Class status changes from "Not Started" → "In Progress"
 - ☐ Score entries 2-4

- ☐ ☒ Class status remains "In Progress"
- ☐ Score final (5th) entry
- ☐ ☒ Class status changes to "Completed"
- ☐ ☒ Final placements automatically calculated (check for 1st, 2nd, 3rd place)

Expected Result: Class completion auto-updates, placements calculated on completion.

1.5 Entry Data Fetching & Caching

Covers: `entryDataLayer.test.ts`, `entryReplication.test.ts`

Test Steps:

1. Log in as Judge
2. Navigate to Class List → Select class with 20+ entries
3. **Cache Performance:**
 - ☐ Open Entry List (first load)
 - ☐ Note load time (~1-2 seconds)
 - ☐ Navigate away, come back to same entry list
 - ☐ ☒ Second load is instant (<100ms from cache)
4. **Offline Cache:**
 - ☐ With entry list open, go offline (DevTools → Network → Offline)
 - ☐ Navigate to different page, return to entry list
 - ☐ ☒ Entry list loads from IndexedDB cache even offline
 - ☐ ☒ All entry data visible (names, breeds, statuses)

Expected Result: Entries load instantly from cache, work offline.

1.6 Real-Time Entry Subscriptions

Covers: `entrySubscriptions.test.ts`

Test Steps:

1. Open two browser windows side-by-side (or two devices)
2. Log into same show with same passcode on both
3. Navigate to same Entry List on both
4. **Real-Time Score Update:**
 - ☐ In Window 1: Score an entry
 - ☐ ☒ Window 2: Entry score appears within 2 seconds
5. **Real-Time Status Update:**
 - ☐ In Window 1: Change entry status to "On Deck"
 - ☐ ☒ Window 2: Status changes to yellow within 2 seconds
6. **Real-Time Delete:**
 - ☐ In Window 1 (admin): Delete an entry
 - ☐ ☒ Window 2: Entry disappears from list

Expected Result: All changes sync in real-time across all connected clients.

1.7 Offline-First Pattern Consistency

Covers: `offline-first-pattern-consistency.test.ts`

Test Steps:

1. Log in as Judge
2. Navigate to Entry List
3. **Pattern Validation:**
 - ☐ Score 3 entries rapidly (within 5 seconds)
 - ☐ ☒ All 3 UI updates happen instantly (<50ms each)
 - ☐ ☒ No loading spinners block user input
 - ☐ ☒ User can continue scoring without waiting
4. **Connection Drop Recovery:**
 - ☐ Start scoring an entry
 - ☐ Go offline mid-score entry (before submission)
 - ☐ Submit score
 - ☐ ☒ Score appears in UI immediately
 - ☐ Go back online
 - ☐ ☒ Score syncs automatically, no data loss

Expected Result: UI always responsive, offline-first pattern consistent.

Section 2: Admin & Settings UI (MEDIUM PRIORITY)

2.1 Admin Name Dialog

Covers: `AdminNameDialog.test.tsx`

Test Steps:

1. Log in with Admin passcode
2. Navigate to Admin → Competition Admin
3. **Set Admin Name:**
 - ☐ Click "Set Admin Name" button
 - ☐ ☒ Dialog opens with text input
 - ☐ Enter name: "Test Admin"
 - ☐ Click "Save"
 - ☐ ☒ Dialog closes
 - ☐ ☒ Name appears in admin interface
 - ☐ Refresh page
 - ☐ ☒ Name persists in localStorage

Expected Result: Admin name saves and persists across sessions.

2.2 Data Management Section (Settings)

Covers: `DataManagementSection.test.tsx`

Test Steps:

1. Navigate to Settings page
2. Scroll to "Data Management" section
3. **Storage Usage:**
 - ☐ ☒ Storage usage displays (e.g., "2.5 MB / 50 MB")
 - ☐ Click "Refresh Usage"
 - ☐ ☒ Numbers update
4. **Clear All Data:**
 - ☐ Click "Clear All Data"
 - ☐ ☒ Confirmation dialog appears
 - ☐ Click "Cancel"
 - ☐ ☒ Nothing cleared, data intact
 - ☐ Click "Clear All Data" again → "Confirm"
 - ☐ ☒ All cached data cleared
 - ☐ ☒ Requires re-login

Expected Result: Data management controls work, clear data requires confirmation.

2.3 Developer Tools Section (Settings)

Covers: `DeveloperToolsSection.test.tsx`

Test Steps:

1. Navigate to Settings → scroll to "Developer Tools"
2. **IndexedDB Diagnostics:**
 - ☐ Click "Run Diagnostics"
 - ☐ ☒ Modal shows database info (tables, record counts)
 - ☐ ☒ All tables listed (cache, mutations, shows, metadata)
3. **Force Sync:**
 - ☐ Click "Force Sync Now"
 - ☐ ☒ Sync starts, progress indicator shown
 - ☐ ☒ Completion message appears
4. **Clear Cache:**
 - ☐ Click "Clear Cache Only"
 - ☐ ☒ Confirmation dialog
 - ☐ Confirm
 - ☐ ☒ Cache cleared but user data intact (no logout)

Expected Result: Developer tools functional, useful for debugging.

2.4 Push Notification Settings

Covers: `PushNotificationSettings.test.tsx`

Test Steps:

1. Navigate to Settings → "Push Notifications" section
2. **Enable Notifications:**
 - ☐ Toggle "Enable Notifications" ON
 - ☐ ☒ Browser permission prompt appears
 - ☐ Click "Allow"
 - ☐ ☒ Toggle remains ON
 - ☐ ☒ Test notification appears (if implemented)
3. **Notification Preferences:**
 - ☐ Toggle "Dog Alerts" ON
 - ☐ Toggle "Urgent Announcements" ON
 - ☐ ☒ Preferences save immediately
 - ☐ Refresh page
 - ☐ ☒ Settings persist
4. **Disable Notifications:**
 - ☐ Toggle "Enable Notifications" OFF
 - ☐ ☒ All sub-options gray out/disable

Expected Result: Notification permissions request correctly, settings persist.

2.5 Voice Settings Section

Covers: `VoiceSettingsSection.test.tsx`

Test Steps:

1. Navigate to Settings → "Voice Announcements"
2. **Enable Voice:**
 - ☐ Toggle "Enable Voice" ON
 - ☐ ☒ Voice options appear
3. **Voice Selection:**
 - ☐ Click "Select Voice" dropdown
 - ☐ ☒ List of available voices appears
 - ☐ Select a voice
 - ☐ Click "Test Voice"
 - ☐ ☒ Test announcement plays in selected voice
4. **Volume & Speed:**
 - ☐ Adjust volume slider
 - ☐ Adjust speed slider
 - ☐ Click "Test Voice" again
 - ☐ ☒ Announcement plays with new volume/speed
5. **Announcement Types:**
 - ☐ Toggle "30-Second Warning" ON
 - ☐ Toggle "Time Expired" OFF
 - ☐ ☒ Settings save
 - ☐ Navigate to active scoresheet with timer

- ☐ ☒ 30-second warning announces when timer hits 30s
- ☐ ☒ No announcement at 0 seconds (disabled)

Expected Result: Voice settings functional, announcements play correctly.

Section 3: Scoresheet Components (LOW PRIORITY)

3.1 Timer Display Component

Covers: `TimerDisplay.test.tsx`

Test Steps:

1. Log in as Judge
2. Open an AKC Scent Work scoresheet
3. **Start Timer:**
 - ☐ Click "Start" button
 - ☐ ☒ Timer counts up in real-time
 - ☐ ☒ Display shows MM:SS.mmm format
4. **Stop Timer:**
 - ☐ Click "Stop" after ~15 seconds
 - ☐ ☒ Timer freezes
 - ☐ ☒ Time value shows in score form
5. **Reset Timer:**
 - ☐ Click "Reset"
 - ☐ ☒ Timer returns to 00:00.000

Expected Result: Timer starts/stops/resets correctly, integrates with scoresheet.

3.2 Area Inputs Component

Covers: `AreaInputs.test.tsx`

Test Steps:

1. Open AKC Scent Work scoresheet (Novice level with 4 areas)
2. **Area Time Inputs:**
 - ☐ Enter "45.5" in Container area
 - ☐ Enter "1:23.4" in Interior area
 - ☐ ☒ Both formats accepted (seconds or MM:SS)
 - ☐ ☒ Values normalize to consistent format
3. **Fault Tracking:**
 - ☐ Check "Fault" for Exterior area
 - ☐ ☒ Area marked with fault indicator
 - ☐ ☒ Total faults count updates
4. **Area Summary:**
 - ☐ Complete all 4 areas
 - ☐ ☒ Total time calculates automatically

- ☐ ☒ Qualification status shows (Q/NQ)

Expected Result: Area inputs accept multiple time formats, calculate totals.

3.3 Nationals Points Display

Covers: `NationalsPointsDisplay.test.tsx`

Test Steps:

1. Open AKC Nationals scoresheet
2. **Points Breakdown:**
 - ☐ Enter performance data
 - ☐ ☒ Point breakdown shows by category:
 - Clear finds
 - False alerts
 - Time bonus
 - Deductions
 - ☐ ☒ Total points calculated correctly
3. **Qualification Threshold:**
 - ☐ Enter score that qualifies (e.g., 85 points)
 - ☐ ☒ "Qualified" indicator shows in green
 - ☐ Enter score below threshold (e.g., 70 points)
 - ☐ ☒ "Not Qualified" shows in red

Expected Result: Points calculate correctly, qualification status accurate.

Section 4: Entry List Persistence (MEDIUM PRIORITY)

4.1 Entry List State Persistence

Covers: `EntryList.persistence.test.tsx`

Test Steps:

1. Log in as Judge
2. Navigate to Entry List for a class
3. **Apply Filters:**
 - ☐ Enter search term "Border"
 - ☐ Change status filter to "Qualified Only"
 - ☐ ☒ Entry list filters
4. **Persistence Check:**
 - ☐ Navigate away (to Home page)
 - ☐ Navigate back to same Entry List
 - ☐ ☒ Search term still "Border"
 - ☐ ☒ Status filter still "Qualified Only"
 - ☐ ☒ Same entries showing
5. **Scroll Position:**

- ☐ Scroll to bottom of long entry list
- ☐ Navigate away and back
- ☐ ☒ Scroll position restored (or near bottom)

Expected Result: Entry list filters and state persist across navigation.

Completion Checklist

Before Deployment, Confirm:

- ☐ **All Section 1 tests passed** (Entry Management - HIGH PRIORITY)
- ☐ **At least 80% of Section 2 passed** (Admin & Settings - MEDIUM)
- ☐ **At least 50% of Section 3 passed** (Scoresheet Components - LOW)
- ☐ **Section 4 persistence test passed** (Entry List)
- ☐ **No critical bugs discovered** during manual testing
- ☐ **Any bugs found logged** in GitHub Issues

Logging Issues

If you find a bug during manual testing:

1. **Severity Classification:**

- Critical:** Blocks core workflow (scoring, login)
- High:** Major feature broken, has workaround
- Medium:** Minor feature issue
- Low:** Cosmetic or rare edge case

2. **Report Format:**

```
**Test**: [Section 1.1 - Score Submission]
**Severity**: Critical
**Steps to Reproduce**:
1. Log in as judge
2. Score entry while offline
3. Go back online

**Expected**: Score syncs automatically
**Actual**: Score lost, entry shows as unscored

**Browser**: Chrome 120
**Notes**: Reproducible 100% of time
```

Test Completion Sign-Off

Tested By: _____ **Date:** _____ **Browser Version:** _____ **Test Environment:** ☐ Local Dev ☐ Staging ☐ Production

Overall Status: ☐ Ready to Deploy ☐ Issues Found (see below)

Issues Found:

1.

2.

3.

Sign-off: ☐ Approved for deployment