

Dentra Dashboard - Comprehensive Test Cases

Test Environment

Dashboard URL: <http://localhost:3001> (or preview URL)

Backend API: <https://c25fdd09e.preview.abacusai.app>

Swagger Docs: <https://c25fdd09e.preview.abacusai.app/api-docs>

Test Case Categories

- A. Visual & UI/UX Testing**
 - B. Functional Feature Testing**
 - C. Data Integration Testing**
 - D. End-to-End Workflow Testing**
 - E. Performance & Error Handling**
-

A. VISUAL & UI/UX TESTING

TC-UI-01: Landing Page Visual Check

Objective: Verify the dashboard loads with proper branding and layout

Steps:

1. Open dashboard URL
2. Observe the header, navigation, and overall layout

Expected Results:

- Dentra logo and “AI Voice Agent” subtitle visible in top-left
 - Navigation tabs: Dashboard | Appointments | Calls | Escalations | Clinics
 - “Dashboard” tab is highlighted/active (blue background)
 - Clinic selector dropdown on the right showing “All Clinics”
 - Clean, modern design with proper spacing and colors
 - No layout breaks, overlapping elements, or visual glitches
-

TC-UI-02: Stats Cards Display

Objective: Verify all 6 stat cards render correctly

Steps:

1. On the Dashboard page, observe the 6 stat cards

Expected Results:

- **Card 1:** “Total Calls” with phone icon (blue), showing count and breakdown
 - **Card 2:** “Appointments” with calendar icon (blue), showing confirmed/cancelled
 - **Card 3:** “Escalations” with alert icon (orange), showing attention count
 - **Card 4:** “Estimated Revenue” with dollar icon (green), showing USD amount
 - **Card 5:** “Success Rate” with checkmark icon (blue), showing percentage
 - **Card 6:** “System Health” with activity icon (purple), showing “Monitor” link
 - All cards have consistent styling, rounded corners, subtle shadows
 - Icons are colorful and properly aligned
-

TC-UI-03: System Health Section**Objective:** Verify system health display and metrics**Steps:**

1. Scroll down to “System Health” section
2. Observe the status badge and metrics

Expected Results:

- Green banner with checkmark showing “Healthy” status
 - Last updated timestamp displayed
 - Four metrics displayed in a grid:
 - CALLS (24H)
 - ERROR RATE
 - ESCALATION RATE
 - AVG DURATION
 - Clean typography and spacing
-

TC-UI-04: Responsive Navigation**Objective:** Verify navigation works and highlights active tab**Steps:**

1. Click “Appointments” tab
2. Click “Calls” tab
3. Click “Escalations” tab
4. Click “Clinics” tab
5. Click “Dashboard” tab to return

Expected Results:

- Each click immediately highlights the clicked tab (blue background)
 - Previous tab returns to default state (gray text)
 - Page content changes to show relevant section
 - No page refresh/reload (SPA behavior)
 - Smooth transitions
-

TC-UI-05: Color Theme & Branding

Objective: Verify consistent color scheme and professional appearance

Expected Results:

- Primary blue color (#3B82F6) used for highlights and active states
 - Green (#10B981) for revenue/positive metrics
 - Orange (#F59E0B) for warnings/escalations
 - Purple (#8B5CF6) for system health
 - Clean white/gray backgrounds
 - Professional, modern sans-serif font (likely Inter or similar)
 - Proper contrast ratios for accessibility
-

B. FUNCTIONAL FEATURE TESTING

TC-FUNC-01: Clinic Filter Functionality

Objective: Test clinic selector dropdown and data filtering

Steps:

1. Click the “Clinic” dropdown (top-right)
2. Observe available options
3. Select “Chicago Dental Center”
4. Observe stat changes
5. Select “NY Premium Dental”
6. Observe stat changes
7. Select “All Clinics”

Expected Results:

- Dropdown shows: All Clinics, Chicago Dental Center, NY Premium Dental, LA Smile Studio, Houston Dental, Phoenix Family Dental
 - Stats update dynamically when a specific clinic is selected
 - “All Clinics” shows aggregated data across all clinics
 - No console errors during filtering
 - Smooth data loading (may show loading state briefly)
-

TC-FUNC-02: Real-Time Stats Display

Objective: Verify dashboard stats show accurate data

Steps:

1. Select “All Clinics”
2. Review each stat card value
3. Compare with backend data (use Swagger to fetch /dashboard/stats)

Expected Results:

- **Total Calls:** Shows count of calls in DB (currently 0)
- **Appointments:** Shows 50 appointments (25 confirmed, 25 cancelled from seed data)
- **Escalations:** Shows count of unresolved escalations
- **Estimated Revenue:** Calculated correctly (should be \$0 if no completed appointments)

- **Success Rate:** Percentage of successful call completions
 - All numbers match backend API responses
-

TC-FUNC-03: Appointments Page

Objective: Test appointments list view and data display

Steps:

1. Click “Appointments” tab
2. Observe the appointments list
3. Check column headers: Patient, Clinic, Service, Date/Time, Status
4. Look for pagination controls if >20 appointments
5. Try filtering by status (if available)

Expected Results:

- List of appointments displayed in a clean table
 - Each row shows: patient name, clinic name, service type, appointment date/time, status badge
 - Status badges colored appropriately:
 - Green for “confirmed”
 - Gray for “cancelled”
 - Yellow/Blue for “pending”
 - Green for “completed”
 - Data sorted by date (most recent first)
 - Pagination works if >20 appointments
-

TC-FUNC-04: Calls Page

Objective: Test calls log view and details

Steps:

1. Click “Calls” tab
2. Observe the calls list (may be empty if no calls made)
3. Check for: Caller, Clinic, Duration, Status, Timestamp columns

Expected Results:

- Calls list displayed in table format
 - If empty, shows appropriate “No calls yet” message
 - Each call shows: phone number, clinic, duration, status, timestamp
 - Status badges: completed (green), failed (red), in-progress (blue)
 - Duration formatted properly (e.g., “2m 34s”)
-

TC-FUNC-05: Escalations Page

Objective: Test escalations queue and resolution

Steps:

1. Click “Escalations” tab
2. Observe escalations list (may be empty)

3. If escalations exist, look for “Resolve” button
4. Check escalation details: reason, clinic, timestamp

Expected Results:

- Escalations displayed with clear priority/urgency indicators
 - Each escalation shows: type, reason, clinic, patient, timestamp
 - “Resolve” button available for each escalation
 - If empty, shows “No escalations - All calls handled successfully!”
 - Escalations sorted by urgency/timestamp
-

TC-FUNC-06: Clinics Page

Objective: Test clinics directory and information display

Steps:

1. Click “Clinics” tab
2. Observe the clinics list
3. Check clinic details: name, address, phone, services

Expected Results:

- All 5 clinics listed:
 - Chicago Dental Center
 - NY Premium Dental
 - LA Smile Studio
 - Houston Dental Care
 - Phoenix Family Dental
 - Each clinic shows: full address, phone number, operating hours
 - Clean card-based layout
 - Icons for phone/location
-

C. DATA INTEGRATION TESTING

TC-DATA-01: Backend API Connectivity

Objective: Verify dashboard successfully connects to backend

Steps:

1. Open browser DevTools (F12) > Network tab
2. Refresh the dashboard
3. Observe API calls being made

Expected Results:

- API calls made to: <https://c25fdd09e.preview.abacusai.app>
- Successful responses (200 OK) for:
 - GET /dashboard/stats
 - GET /dashboard/appointments
 - GET /dashboard/health
 - GET /clinics
- No CORS errors
- Response times < 2 seconds

TC-DATA-02: Stats Calculation Accuracy

Objective: Verify revenue and success rate calculations

Steps:

1. Use Swagger docs to fetch raw data:
 - GET /dashboard/appointments?status=completed
 - GET /dashboard/stats
2. Manually calculate expected revenue
3. Compare with dashboard display

Expected Results:

- Revenue = Sum of (appointment service cost) for all completed appointments
 - Success Rate = (completed calls / total calls) * 100
 - Dashboard matches manual calculations
 - Proper currency formatting (\$1,234.56)
-

TC-DATA-03: Real-Time Data Updates

Objective: Test if dashboard updates when backend data changes

Steps:

1. Note current stats on dashboard
2. Use Swagger to create a new appointment (POST /dashboard/appointments)
3. Refresh dashboard
4. Observe updated stats

Expected Results:

- Appointment count increases by 1
 - New appointment appears in Appointments list
 - Stats update correctly
 - No stale data displayed
-

TC-DATA-04: Empty State Handling

Objective: Verify proper handling when no data exists

Steps:

1. Select a clinic with no appointments/calls
2. Observe each section

Expected Results:

- Stats show "0" gracefully (not errors)
- Lists show friendly empty messages:
 - "No appointments scheduled yet"
 - "No calls recorded"
 - "No escalations - great job!"
- No "undefined" or error messages
- UI remains clean and professional

D. END-TO-END WORKFLOW TESTING

TC-E2E-01: New Appointment Workflow

Objective: Test complete appointment booking flow visibility

Steps:

1. Start on Dashboard (All Clinics view)
2. Note current appointment count
3. Make a simulated call via Twilio webhook (POST /webhook/voice)
4. Follow voice agent flow to book appointment
5. Return to dashboard
6. Verify new appointment appears

Expected Results:

- Dashboard appointment count increases
 - New appointment visible in Appointments tab
 - Patient info displayed correctly
 - Appointment status is “confirmed”
 - Revenue estimate updates (if completed)
-

TC-E2E-02: Escalation Workflow

Objective: Test escalation creation and resolution

Steps:

1. Create an escalation (via API or simulated call with complex issue)
2. Check Dashboard - Escalation count should increase
3. Go to Escalations tab
4. Click “Resolve” on the escalation
5. Verify escalation removed from queue

Expected Results:

- Escalation appears in dashboard stats immediately
 - Escalation details visible in Escalations tab
 - “Resolve” button works
 - After resolution, count decreases
 - Resolved escalation no longer in queue
-

TC-E2E-03: Multi-Clinic Management

Objective: Test managing multiple clinics simultaneously

Steps:

1. View “All Clinics” aggregated stats
2. Filter to “Chicago Dental Center” - note stats
3. Filter to “NY Premium Dental” - note stats
4. Sum up individual clinic stats manually
5. Compare with “All Clinics” totals

Expected Results:

- All Clinics stats = sum of individual clinic stats
 - Each clinic's data is isolated correctly
 - No data leakage between clinics
 - Filtering is fast and responsive
-

TC-E2E-04: System Health Monitoring**Objective:** Test real-time system health visibility**Steps:**

1. Observe System Health showing "Healthy"
2. Make several API calls to increase call count
3. Refresh dashboard
4. Verify metrics update

Expected Results:

- CALLS (24H) shows accurate count
 - ERROR RATE calculates correctly (failed/total)
 - ESCALATION RATE shows escalations/total
 - AVG DURATION shows average call time
 - Status changes to "Warning" or "Critical" if thresholds exceeded
-

E. PERFORMANCE & ERROR HANDLING**TC-PERF-01: Page Load Performance****Objective:** Verify dashboard loads quickly**Steps:**

1. Open dashboard with DevTools > Performance
2. Measure initial load time
3. Measure time to interactive

Expected Results:

- Initial page load < 2 seconds
 - API data loaded and displayed < 3 seconds
 - No layout shift during loading
 - Smooth rendering (60 FPS)
-

TC-PERF-02: Filter Response Time**Objective:** Test clinic filter performance**Steps:**

1. Switch between different clinics rapidly
2. Observe response time

Expected Results:

- Filter changes reflected < 1 second
 - No lag or freezing
 - Loading indicators shown if needed
 - Smooth transitions
-

TC-ERROR-01: Backend Offline Handling**Objective:** Test behavior when backend is unavailable**Steps:**

1. Stop the backend service
2. Refresh dashboard
3. Observe error handling

Expected Results:

- Friendly error message displayed
 - No cryptic error codes shown to user
 - Suggestion to “Check back later” or “Contact support”
 - UI doesn’t break or show blank page
 - Console logs errors for debugging
-

TC-ERROR-02: Invalid Data Handling**Objective:** Test handling of malformed API responses**Steps:**

1. Use Swagger to create invalid data (if possible)
2. Observe dashboard behavior

Expected Results:

- Dashboard handles missing fields gracefully
 - Shows “N/A” or “—” for missing data
 - No console errors or crashes
 - Invalid dates/times formatted safely
-

TC-ERROR-03: Network Timeout Handling**Objective:** Test behavior with slow network**Steps:**

1. Throttle network in DevTools (Slow 3G)
2. Refresh dashboard
3. Observe loading behavior

Expected Results:

- Loading indicators shown
- Data loads eventually (within timeout)

- Timeout errors handled gracefully
 - Retry mechanism works
-

QUICK SMOKE TEST CHECKLIST

Use this for rapid validation:

Visual Check (30 seconds)

- [] Dashboard loads without errors
- [] All 6 stat cards visible
- [] Navigation tabs work
- [] No visual glitches

Data Check (1 minute)

- [] Stats show real numbers (50 appointments, etc.)
- [] Clinic filter works
- [] Appointments tab shows data
- [] System health displays

Interaction Check (1 minute)

- [] Click all nav tabs - each loads
- [] Clinic dropdown filters data
- [] No console errors
- [] Smooth performance



EXPECTED SEED DATA

Your database should have:

- **5 Clinics:** Chicago, NY, LA, Houston, Phoenix
- **20 Patients:** Various names and contact info
- **25 Services:** Cleanings, root canals, crowns, etc. with prices
- **50 Appointments:** Mix of confirmed/cancelled/pending/completed
- **0 Calls:** No calls made yet (test this feature separately)
- **0 Escalations:** Clean slate for escalation testing



KNOWN ISSUES TO WATCH FOR

1. **Revenue showing \$0:** Expected if no “completed” appointments exist
2. **Success Rate 0.0%:** Expected if no calls made yet
3. **Total Calls 0:** Normal - no inbound calls processed yet
4. **Appointments showing 50:** Correct - matches seed data

SUCCESS CRITERIA

The dashboard is considered **fully functional** if:

- All pages load without errors
 - Stats display accurate data from backend
 - Clinic filtering works correctly
 - Navigation is smooth and responsive
 - UI is clean, modern, and professional
 - No console errors in DevTools
 - Data updates when backend changes
 - Empty states handled gracefully
-

TESTING PRIORITY

HIGH PRIORITY (Must Pass):

- TC-UI-01, TC-UI-02, TC-FUNC-01, TC-FUNC-02, TC-DATA-01

MEDIUM PRIORITY (Should Pass):

- TC-FUNC-03, TC-FUNC-04, TC-FUNC-05, TC-DATA-02, TC-E2E-01

LOW PRIORITY (Nice to Have):

- TC-PERF-01, TC-ERROR-01, TC-ERROR-02
-

TEST RESULT TEMPLATE

For each test case:

Test ID: TC-XXX-XX

Status: PASS / FAIL / PARTIAL

Notes: [Any observations, screenshots, or issues]

Browser: [Chrome/Firefox/Safari]

Date: [Test date]

THE MODERN UI YOU'RE TESTING

What makes this dashboard special:

1. **Clean Minimalism:** No clutter, every element has purpose
2. **Colorful Icons:** Blue, green, orange, purple for visual hierarchy
3. **Smooth Interactions:** Hover effects, transitions, responsive feedback
4. **Data Visualization:** Clear metrics, easy to scan
5. **Professional Typography:** Consistent fonts, proper hierarchy
6. **Responsive Layout:** Works on all screen sizes
7. **Real-Time Updates:** Live data from production backend
8. **Intuitive Navigation:** One click to any section

9. **Status Indicators:** Color-coded badges for quick status recognition

10. **System Monitoring:** Built-in health dashboard

This is a **production-grade dashboard** ready for real dental clinics! 🦷🌟

Happy Testing! Let me know which test cases you'd like to focus on first, or if you encounter any issues during testing. 🚀