# Preview and Testing

## Overview

Preview mode allows you to test your assessment as end users will experience it. You can verify conditional logic, test triggered questions, and ensure the assessment flows correctly. This section covers testing workflows and validation strategies.

## Understanding Preview Mode

### Edit Mode vs. Preview Mode

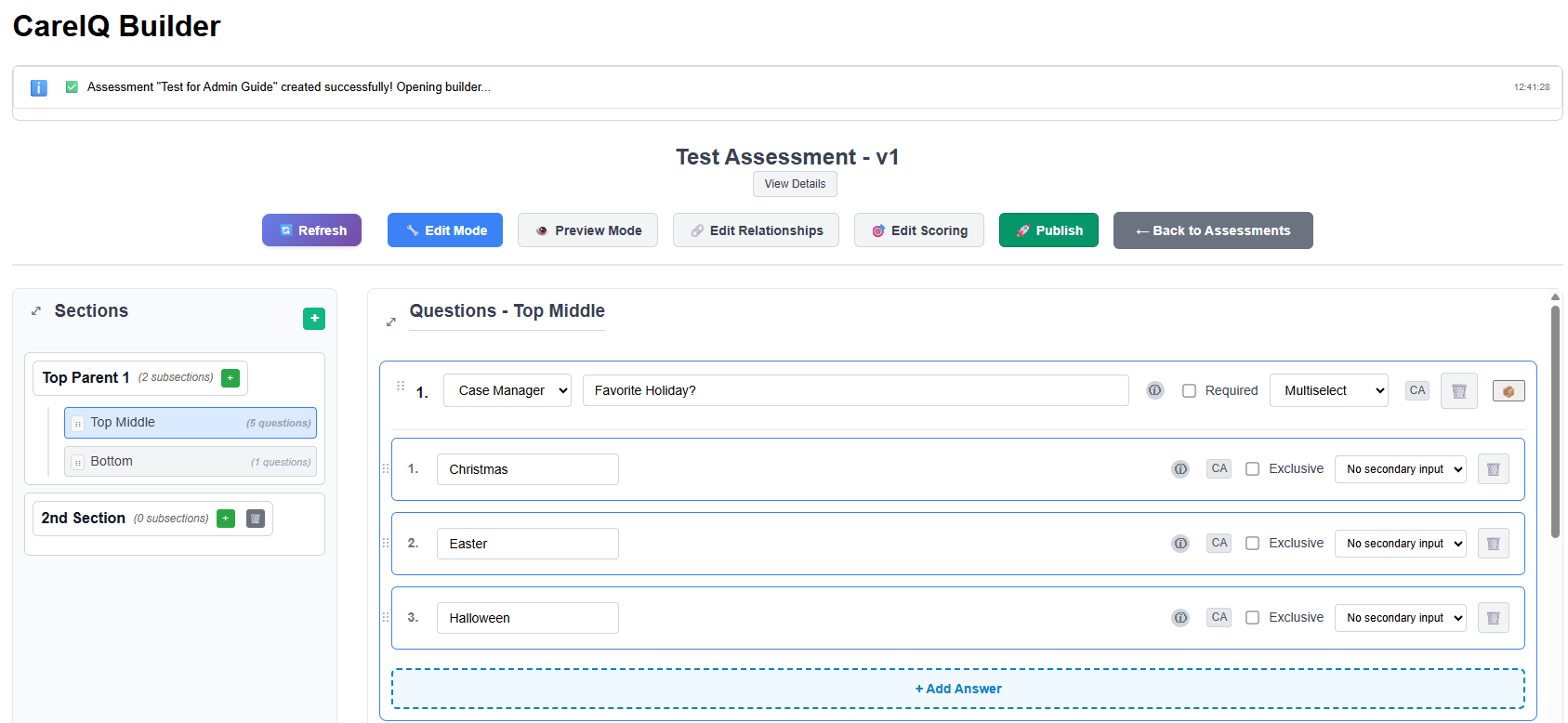
CareIQ Builder has two distinct modes:

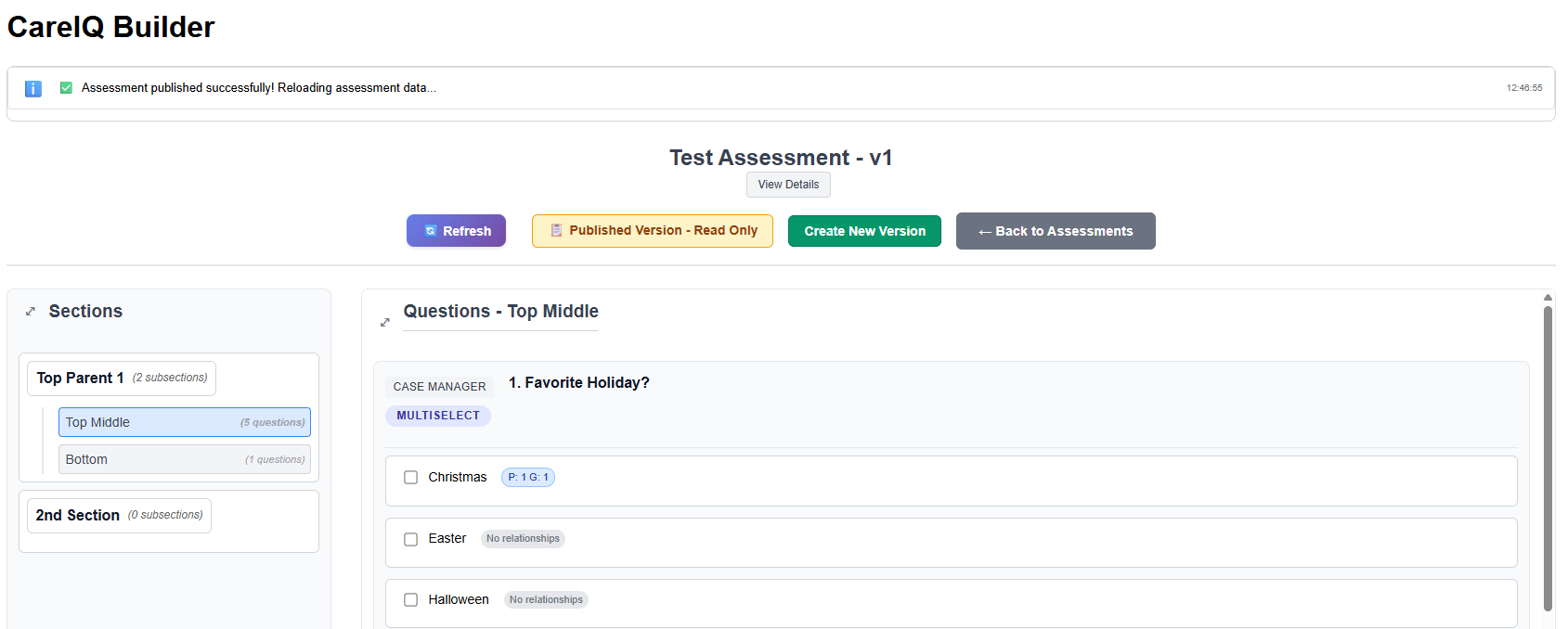
#### **Edit Mode**

* Full editing capabilities
* Can create, modify, and delete content
* Shows all controls and buttons
* All questions visible regardless of conditional logic
* Save/Cancel buttons appear for unsaved changes
* Used for building and modifying assessments

#### **Preview Mode**

* Read-only view
* Simulates end-user experience
* Shows only relevant questions based on selections
* Conditional logic is active
* No edit controls visible
* Used for testing and validation





### Switching Between Modes

#### Entering Preview Mode

1. **Locate Mode Toggle**
   * In the top bar of the builder interface
   * Look for **“Preview Mode”** button or toggle
2. **Click Preview Mode**
   * Button appearance changes
   * Edit controls disappear
   * Assessment displays as users will see it

**A screenshot of a computer

AI-generated content may be incorrect.**

#### Returning to Edit Mode

1. **Locate Mode Toggle**
   * Same button in top bar
   * Now shows **“Edit Mode”** label
2. **Click Edit Mode**
   * Edit controls reappear
   * All questions become visible
   * Can resume editing

### What Changes in Preview Mode?

When you enter Preview mode:

**Visible Changes:** - Add/Edit/Delete buttons disappear - Section management controls hide - Question management controls hide - All questions show (initially) - Answer selections become active/clickable

**Functional Changes:** - Can select answers (radio buttons, checkboxes) - Triggered questions appear/disappear based on selections - Conditional logic is active - Cannot modify content - Cannot add relationships

## Testing Basic Questions

### Single Select Questions

1. **Navigate to Section**
   * Click section in left panel
   * Questions display in main area
2. **View Question**
   * Question text displays
   * Radio button options appear

**A close up of a person

AI-generated content may be incorrect.**

1. **Select an Answer**
   * Click a radio button
   * Selection is highlighted
   * Only one option can be selected at a time
2. **Change Selection**
   * Click a different radio button
   * Previous selection deselects automatically

### Multiselect Questions

1. **View Question**
   * Question text displays
   * Checkbox options appear
2. **Select Multiple Answers**
   * Click checkboxes to select
   * Multiple selections allowed
   * Each selection remains checked

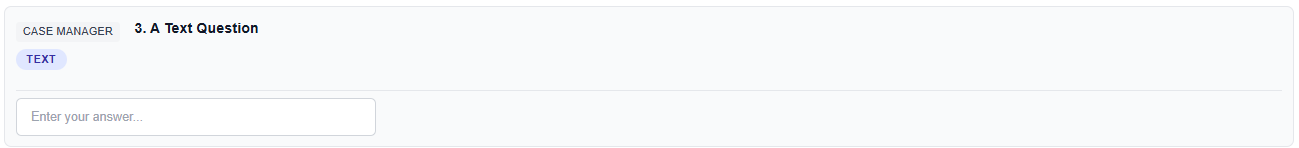
**A white rectangular object with a black border

AI-generated content may be incorrect.**

1. **Test Mutually Exclusive Answers**
   * If an answer is marked mutually exclusive (e.g., “None of the above”):
   * Select regular answers first
   * Then select the mutually exclusive answer
   * All other selections should clear
   * Only the mutually exclusive answer remains selected

### Free Text Questions

1. **View Question**
   * Question text displays
   * Text input field appears
2. **Enter Text**
   * Click in the text field
   * Type a response
   * Text should accept input normally

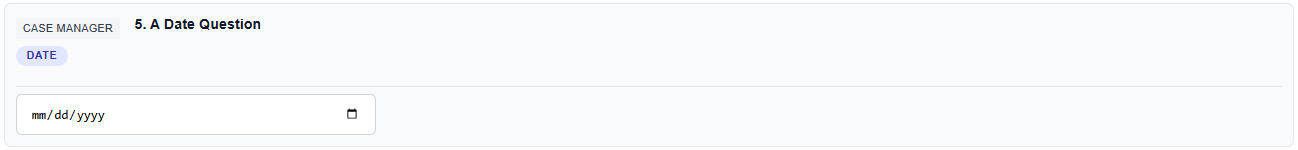
****

### Numeric Questions

1. **View Question**
   * Question text displays
   * Numeric input field appears
2. **Enter Number**
   * Click in the field
   * Type a numeric value
   * Field should accept only numbers
3. **Test Validation**
   * Try entering text (should be prevented or show error)
   * Try negative numbers (if applicable)
   * Try decimals (if applicable)

### Date Questions

1. **View Question**
   * Question text displays
   * Date picker appears
2. **Select Date**
   * Click the date field or calendar icon
   * Date picker opens
   * Select a date
   * Date populates in field

****

## Testing Conditional Logic

### Understanding Triggered Questions

Triggered questions (conditional questions) appear only when specific answers are selected. Testing this logic is critical.

### Basic Triggered Question Test

**Scenario:** Question B should appear only when Answer A1 is selected

1. **Navigate to Section**
   * Go to section containing Question A
2. **Initial State**
   * Question A is visible
   * Question B is NOT visible (hasn’t been triggered)
3. **Select Triggering Answer**
   * Select Answer A1
   * Question B should appear below Question A

Before

A screenshot of a computer

AI-generated content may be incorrect.

After

A screenshot of a chat

AI-generated content may be incorrect.

1. **Verify Question B Appears**
   * Confirm Question B is now visible
   * Verify it’s in the correct position
   * Check that question text is correct
2. **Deselect Triggering Answer**
   * Select a different answer for Question A
   * Question B should disappear
3. **Re-test**
   * Select Answer A1 again
   * Question B should reappear
   * Repeat several times to ensure consistency

### Multi-Level Conditional Testing

**Scenario:** Question C is triggered by an answer to Question B, which itself is triggered by Question A

1. **Start Fresh**
   * Ensure no answers are selected
   * Only Question A should be visible
2. **Trigger First Level**
   * Select answer that triggers Question B
   * Verify Question B appears
   * Question C should NOT appear yet
3. **Trigger Second Level**
   * Select answer in Question B that triggers Question C
   * Verify Question C appears
   * All three questions should now be visible

**A screenshot of a computer

AI-generated content may be incorrect.**

1. **Test Backward Collapse**
   * Change answer in Question B to one that doesn’t trigger C
   * Question C should disappear
   * Question B should remain
2. **Test Full Collapse**
   * Change answer in Question A to one that doesn’t trigger B
   * Both Question B and C should disappear
   * Only Question A remains

### Multiple Trigger Sources

**Scenario:** Question X is triggered by Answer A1 OR Answer B1

1. **Test First Trigger**
   * Select Answer A1
   * Question X should appear
2. **Test Second Trigger**
   * Deselect Answer A1 (Question X disappears)
   * Select Answer B1
   * Question X should appear again
3. **Test Both Triggers**
   * Select both Answer A1 and Answer B1
   * Question X should remain visible
   * Deselect one, X should still show
   * Deselect both, X should disappear

## Testing Secondary Inputs

### Text Secondary Input

**Scenario:** Answer has “Other (please specify)” with text input

1. **Select the Answer**
   * Click the answer with secondary input
   * Text input field should appear

**Screenshot Placeholder 58**: Answer with secondary text input

1. **Enter Text**
   * Click in the secondary input field
   * Type text
   * Verify text is accepted
2. **Deselect Answer**
   * Select a different answer
   * Secondary input field should disappear
   * Re-select to verify field reappears

### Date Secondary Input

1. **Select the Answer**
   * Click answer with date secondary input
   * Date picker should appear
2. **Select Date**
   * Choose a date
   * Verify date populates
3. **Change Date**
   * Modify the date selection
   * Verify changes are accepted

### Numeric Secondary Input

1. **Select the Answer**
   * Click answer with numeric secondary input
   * Number field should appear
2. **Enter Number**
   * Type a numeric value
   * Verify only numbers are accepted
3. **Test Validation**
   * Try entering non-numeric characters
   * Should be prevented or show error

## Testing Question Visibility

### Section-Level Visibility

1. **Navigate Between Sections**
   * Click different sections in left panel
   * Verify correct questions display for each section
   * Ensure no questions appear in wrong sections
2. **Test All Sections**
   * Go through each section systematically
   * Verify question count matches Edit mode
   * Check for missing or duplicate questions

### Answer-Based Visibility

1. **Test Each Answer**
   * For questions with multiple answers, test each one
   * Verify correct triggered questions appear
   * Ensure no unexpected questions appear
2. **Cross-Section Triggers**
   * If answer in Section A triggers question in Section B:
   * Select answer in Section A
   * Navigate to Section B
   * Verify triggered question appears

## Creating Test Scenarios

### Develop Test Cases

Before testing, create a structured test plan:

#### **Example Test Case Template:**

Test Case: Diabetes Screening Flow  
Prerequisites: Assessment in Preview mode  
  
Steps:  
1. Navigate to "Medical History" section  
2. Select answer "Yes" to "Have you been diagnosed with diabetes?"  
3. Verify "When were you diagnosed?" question appears  
4. Enter date in date field  
5. Verify "What type of diabetes?" question appears  
6. Select "Type 2"  
7. Verify "Current medications" question appears  
  
Expected Results:  
- All triggered questions appear in sequence  
- Questions appear in correct order  
- All inputs accept data correctly  
- No unexpected questions appear  
  
Actual Results: [Fill in during testing]  
Pass/Fail: [Mark after testing]

### Comprehensive Test Matrix

Create a matrix for complex conditional logic:

| Question | Answer | Should Trigger | Should NOT Trigger |
| --- | --- | --- | --- |
| Q1: Smoker? | Yes | Q2: How much? | Q3: Ever smoked? |
| Q1: Smoker? | No | Q3: Ever smoked? | Q2: How much? |
| Q1: Smoker? | Former | Q4: When quit? | Q2: How much? |

**Screenshot Placeholder 59**: Test matrix example document

### Edge Case Testing

Test unusual or boundary conditions:

1. **Rapid Selection Changes**
   * Quickly select and deselect answers
   * Verify triggered questions appear/disappear correctly
   * Check for any lag or display errors
2. **All Answers Selected (Multiselect)**
   * Select all checkboxes
   * Verify all triggered questions appear
   * Check for performance issues
3. **No Answers Selected**
   * Leave questions unanswered
   * Navigate between sections
   * Verify no errors occur
4. **Maximum Text Length**
   * In free text fields, enter very long text
   * Verify field handles it appropriately
   * Check for truncation or scroll

## Testing Across Sections

### Navigation Testing

1. **Complete Section A**
   * Select answers in Section A
   * Note any triggered questions
2. **Navigate to Section B**
   * Click Section B
   * Verify your selections in Section A remain
3. **Return to Section A**
   * Navigate back to Section A
   * Verify all selections are still present
   * Verify triggered questions still show

### Cross-Section Dependencies

**Scenario:** Answer in Section A affects questions in Section B

1. **Select Answer in Section A**
   * Choose answer that triggers question in different section
2. **Navigate to Affected Section**
   * Go to Section B
   * Verify triggered question appears
3. **Return and Change Answer**
   * Go back to Section A
   * Change the answer selection
   * Return to Section B
   * Verify triggered question disappeared

## Testing PGI Integrations

### Problem-Goal-Intervention Display

While PGI relationships primarily affect backend care planning, you can verify the associations:

1. **Select Answer with PGI Links**
   * Choose answer that has PGI relationships
2. **Check Indicators**
   * Some implementations show indicators
   * May display linked problems or goals
   * Verify correct associations display (if visible in preview)
3. **Test Multiple PGI Answers**
   * Select multiple answers with PGI links
   * Verify all associations are captured

## Performance Testing

### Load Time

1. **Navigate to Large Sections**
   * Sections with many questions (20+)
   * Verify reasonable load time (< 2 seconds)
2. **Trigger Multiple Questions**
   * Select answer that triggers many questions
   * Verify questions appear quickly
   * No noticeable lag
3. **Rapid Navigation**
   * Quickly click through sections
   * Verify smooth transitions
   * No frozen UI or loading issues

### Browser Testing

Test in multiple browsers:

1. **Chrome**
   * Verify all functionality works
   * Check display and formatting
2. **Firefox**
   * Test same scenarios
   * Compare behavior to Chrome
3. **Edge**
   * Repeat tests
   * Note any differences
4. **Safari** (if available)
   * Complete basic test suite
   * Verify compatibility

## Validation Checklist

### Pre-Publication Testing

Before publishing an assessment, complete this checklist:

#### **Content Validation**

* ☐ All questions display correctly
* ☐ All answer options are visible
* ☐ Question types are correct (Single Select, Multiselect, etc.)
* ☐ Question voices are appropriate
* ☐ Tooltips display properly

#### **Conditional Logic**

* ☐ All triggered questions appear when expected
* ☐ Triggered questions hide when triggering answer deselected
* ☐ Multi-level conditional logic works correctly
* ☐ No circular dependencies cause errors
* ☐ Cross-section triggers work

#### **Input Validation**

* ☐ Text fields accept text
* ☐ Numeric fields accept only numbers
* ☐ Date fields open date picker
* ☐ Secondary inputs appear and function
* ☐ Mutually exclusive answers work correctly

#### **Navigation**

* ☐ All sections are accessible
* ☐ Sections contain correct questions
* ☐ Navigation between sections preserves selections
* ☐ No questions in wrong sections

#### **Relationships**

* ☐ Triggered questions are correctly configured
* ☐ Guidelines are linked appropriately
* ☐ PGI relationships are established (verify in Edit mode)
* ☐ Barriers are linked where needed

#### **User Experience**

* ☐ Question flow is logical
* ☐ Assessment completes in reasonable time
* ☐ Instructions are clear
* ☐ No confusing or ambiguous questions
* ☐ Performance is acceptable (no lag)

**Screenshot Placeholder 60**: Validation checklist example

## Common Testing Issues

### Triggered Question Not Appearing

**What to check:** 1. Verify you’re in Preview mode (not Edit) 2. Confirm relationship is configured correctly (check in Edit mode) 3. Ensure triggering answer is actually selected 4. Check that triggered question is in same assessment 5. Refresh page and try again

### Question Appearing When It Shouldn’t

**What to check:** 1. Review all relationships for that question 2. Check if multiple answers trigger it 3. Verify you haven’t selected a triggering answer accidentally 4. Look for unintended relationship configurations 5. Check for duplicate relationships

### Selections Not Persisting

**What to check:** 1. This is expected in Preview mode (not saved) 2. Preview mode is for testing only, not data collection 3. Selections persist during current session 4. Refreshing page clears all selections

### Performance Issues

**What to check:** 1. Browser console for JavaScript errors 2. Network tab for slow API calls 3. Number of questions in section (may need to split) 4. Complexity of conditional logic 5. Browser extensions interfering

### Display Issues

**What to check:** 1. Browser zoom level (should be 100%) 2. Window size (minimum 1400px width recommended) 3. Browser compatibility (use supported browser) 4. Clear browser cache 5. Check for CSS conflicts

## Testing Best Practices

### Test Early and Often

* Don’t wait until assessment is complete
* Test each section as you build it
* Verify conditional logic immediately after configuring
* Catch issues early when they’re easier to fix

### Test All Paths

* Don’t just test the “happy path”
* Test every answer option
* Test every triggered question
* Test all conditional combinations

### Document Issues

* Keep notes of problems found
* Screenshot unexpected behavior
* Record steps to reproduce issues
* Share findings with team

### User Acceptance Testing

* Have actual users test the assessment
* Observe them completing it
* Note confusion points
* Gather feedback on clarity and flow
* Make improvements based on user experience

### Regression Testing

* After making changes, re-test existing functionality
* Ensure fixes don’t break previously working features
* Keep a test script to ensure consistency
* Test all relationships after editing questions or answers

## Tips and Tricks

### Quick Toggle

* Learn the keyboard shortcut for mode toggle (if available)
* Quickly switch to test a change and return to editing

### Browser Developer Tools

* Use browser console to check for errors
* Network tab shows API call timing
* Can help diagnose performance issues

### Split Screen Testing

* Edit mode in one browser window
* Preview mode in another
* Make changes in edit, refresh preview to test

### Test with Real Data

* Use realistic answers
* Enter data as actual users would
* Test with edge cases (very long text, unusual dates)

### Collaborative Testing

* Have colleagues test independently
* Compare results
* Different users may discover different issues
* Fresh eyes catch things you might miss

## Next Steps

After thorough testing: - Make any necessary corrections in **Edit mode** - Re-test after changes - Complete final validation checklist - Proceed to **publishing** when assessment is ready - Consider creating **documentation** of question flow for users