SPEC: Nested Conditions

Spec Status	Complete
GFX Status	complete: STYLES: NESTED CONDITIONS
Screens	complete: STYLES: NESTED CONDITIONS Interactions: ADDITONAL INTERACTIONS:
Dev Status	

PURPOSE

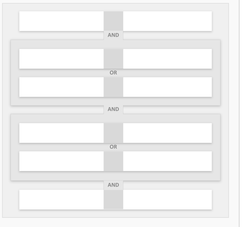
As a user creating/editing a rule

- I want to clearly see conditioned that are grouped (parentheses)
- I want to easily re-order and copy
- I want to create a sub-set of conditions (parentheses)
- I want to combine conditions with OR or AND logic
- I want to make complex logic by creating nested logic e.g. "(A or B) and C"

DESIGN

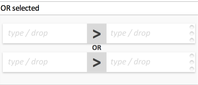
Style

- [650571] There are only 2 tones used for grouped conditions (2 tones of grey shown below). The tones are alternated with each level of nesting.
- [650571] The outermost nesting must always be the same color (i.e. the existing outer nesting doesn't change color when a new nesting is added).

2 levels of nesting	2 grouped conditions next to each other
	

INTERACTIONS

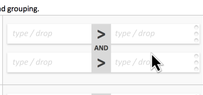
Logic Button

AND selected	OR selected
	
[650571] The conditions appear 'joined'	[650571] The conditions appear 'separated'

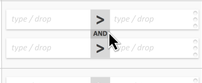
Logic Drop Down

This allows changing the logic and grouping.

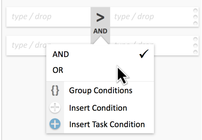
Normal



Hover (slight change of color, no space to show a down arrow)



Drop down options:



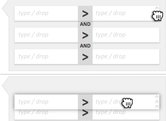
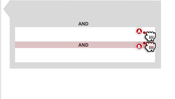
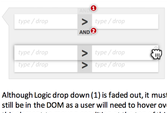
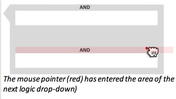
- [650571] Group Conditions - will group the 2 conditions either side of the logic button into a nested group.
  If the 2 conditions either side of the logic button are already in a group (which contains only those 2 conditions) then, and only then will the text read "Ungroup Conditions", and clicking on the button will remove the grouping around them.
  Grouping and Ungrouping are animated [ANIMATION REQUIRED].
- [650571] Insert Condition xxx - these command buttons to insert conditions may include the basic, task or other types of condition - depending on which rules designer is being used. The ANIMATION: Space then Content should be used when adding the inserted condition.

Grouping

Conditions groups can be built in 2 ways:

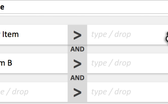
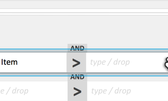
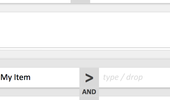
1) [650571] Dragging a condition on top of another condition (and dropping)
2) [650571] Using the logic drop down to group the 2 conditions either side of the logic.

Dragging Conditions - Normal

Description	Wireframe	Diagram
[650571] User Starts Dragging a condition block from the grip to the right of a condition block.		
• [628516] While Dragging, a condition has a drop-shadow. • [664729] Once a drag is in progress, the mouse can move horizontally, but the condition only moves vertically. • [628516] The ONLY time a condition moves horizontally with the mouse is when the mouse cursor exits the outermost condition container (the one connected to the IF socket). (See Diagram) • [628516] A condition can be dropped on the IF part of the if then else block. In this case the condition is added to the end of the condition container block		
• [664729] As the mouse moves down with the dragged condition, the new gap *does not* open when the mouse is over a condition (A). • [664729] The next gap only shows when the mouse is over a 'logic drop down' (B). *Note: The condition being dragged does not show in this diagram.*		
Reserved Space Dragging • [664729] The previous gap stays open until the mouse cursor is within the area of the next logic drop down (hover area shown on right), at which point the gap above will open at the same time as the old gap shuts. • [650571], [664729] Any "Logic Drop Down" that finds itself at the top or bottom of a condition container, must fade out. (there are no Logic DD on the inside edge of a condition container). • [650571], [664729] A new "Logic drop down" fades in at the top of the newly open gap. This should be the same type as the bottom of the open gap (i.e. both are ANDs).	 Although Logic drop down (1) is faded out, it must still be in the DOM as a user will need to hover over this element to move a condition at the top of this condition container.	 *The mouse pointer (red) has entered the area of the next logic drop-down)*

Dragging Conditions - Make a New Group

Hovering over another condition briefly while dragging, will indicate that the condition can be dropped on to create a group.

Description	Wireframe	Diagram
User Starts Dragging a condition block from the grip to the right of a condition block.		Not needed.
[650571] Hovering over another condition still* for 0.5s shows the highlight on the underlying condition. [650571] *'still' means that a timer starts for the 0.5secs on enter, but the mouse mustn't move more than 10px vertically during those 0.5 secs to be considered a hover action. If the mouse moves >10px, the timer starts again.		Not needed.
[650571] The new container is created pushing the outer condition container wider. [MEDIUM ANIMATION REQUIRED] [664729] All of the conditions stay centered. [664729] When the other conditions move up and down to get out of the way of the dragged conditions, any "AND" or "OR" buttons that end up at the edge of a condition container must fade out (see spec for animation) [664729] When dragging a condition vertically, the AND and OR buttons may need to fade in and out so that when you drop the condition, there isn't a glitch, or double AND/OR buttons above or below the dropped condition - see spec.		Not needed.

ADDITONAL INTERACTIONS:

Details	Link
Using the logic dropdown	
Creating a group with dropdown command	STORYBOARD: Grouping with dropdown
Removing a group with dropdown command	STORYBOARD: Ungrouping with dropdown
Using Drag	
★ Overview- (details and order of animations for dragging)	Dragging Nested Conditions Overview
Moving a condition, moving nested conditions	STORYBOARD: Moving Single and Nested Conditions
Creating a group, creating a group within a group	STORYBOARD: Grouping with Drag
Removing a group, removing a group from within a group	STORYBOARD: Ungrouping with Drag

QUESTIONS

? JL: need the bounds for the mouse area [628516] Once a drag is in progress, the mouse

? SJ: More info needed on what happens to Logic when condition is moved:

eg. If you drag A between B &C in the below statement:
A || B & C becomes: B & A & C ?

What happens if you drag A to the bottom of this stack when B and C are grouped? Does the logic stay the same?
A || (B & C) becomes: (B & C) || A ?

What if they're not grouped? Either way the logic is now changed.
A || B & C becomes: B & C & A or B & C || A ?

Are we allowing the user to change the logic by dragging or only the position?
Is AND the default logic for new conditions dragged from toolbox?
Does the condition lose or retain it's class when dragged?