



# Rituparna Saha

*Sr. UX Designer*



Rituparna Saha



I'm Rituparna, an astute Certified Usability Analyst (CUA)™ from HFI (Human Factor International) and Certified in Design Thinking and Innovation Management from the Indian Institute of Management (Kozhikode), with expertise in building and designing UX Strategies.

Over the past 10 years, I've created the product and design solutions for clients across various domains like Enterprise Active Invoices with compliance, Enterprise Business Validation System, Enterprise Active intelligence, Enterprise Intelligent Web Forms, Enterprise Advance shipment Notice, Banking & Finance, Real Estate, Travel & Hospitality, Manufacturing & Logistics, E-learning.

Closely worked with the Business and Development team to create enriched web applications and products. I have learned to identify and incorporate consumer needs into user screens using my creative skills in the best possible way to develop complex things into an easy and creative way that speaks to the end user's goals by embracing new technologies.

I am an individual contributor and manage multiple products. I love giving back to the design community by mentoring designers, sharing design best practices, take an active role in design workshops. Well acquainted with agile methodology.



## Key Clients

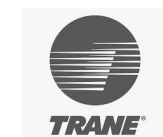
**opentext™** | Active Invoices with Compliance

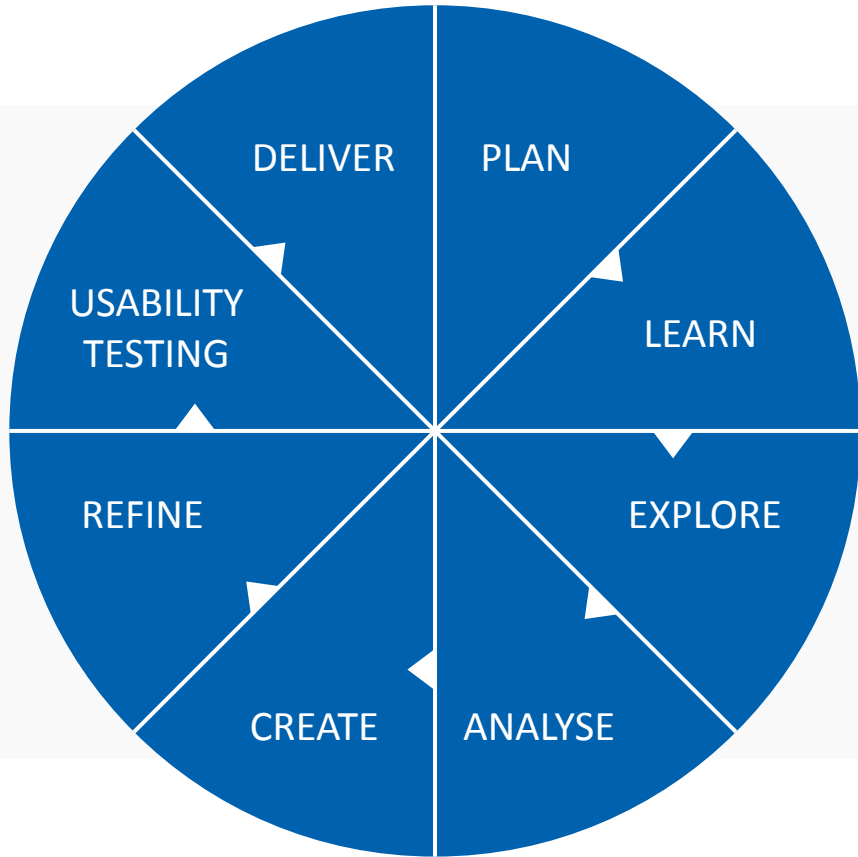
**opentext™** | Business Validation Services

**opentext™** | Intelligent Web Forms

**opentext™** | Advance Shipment Notice

**opentext™** | Active Intelligence





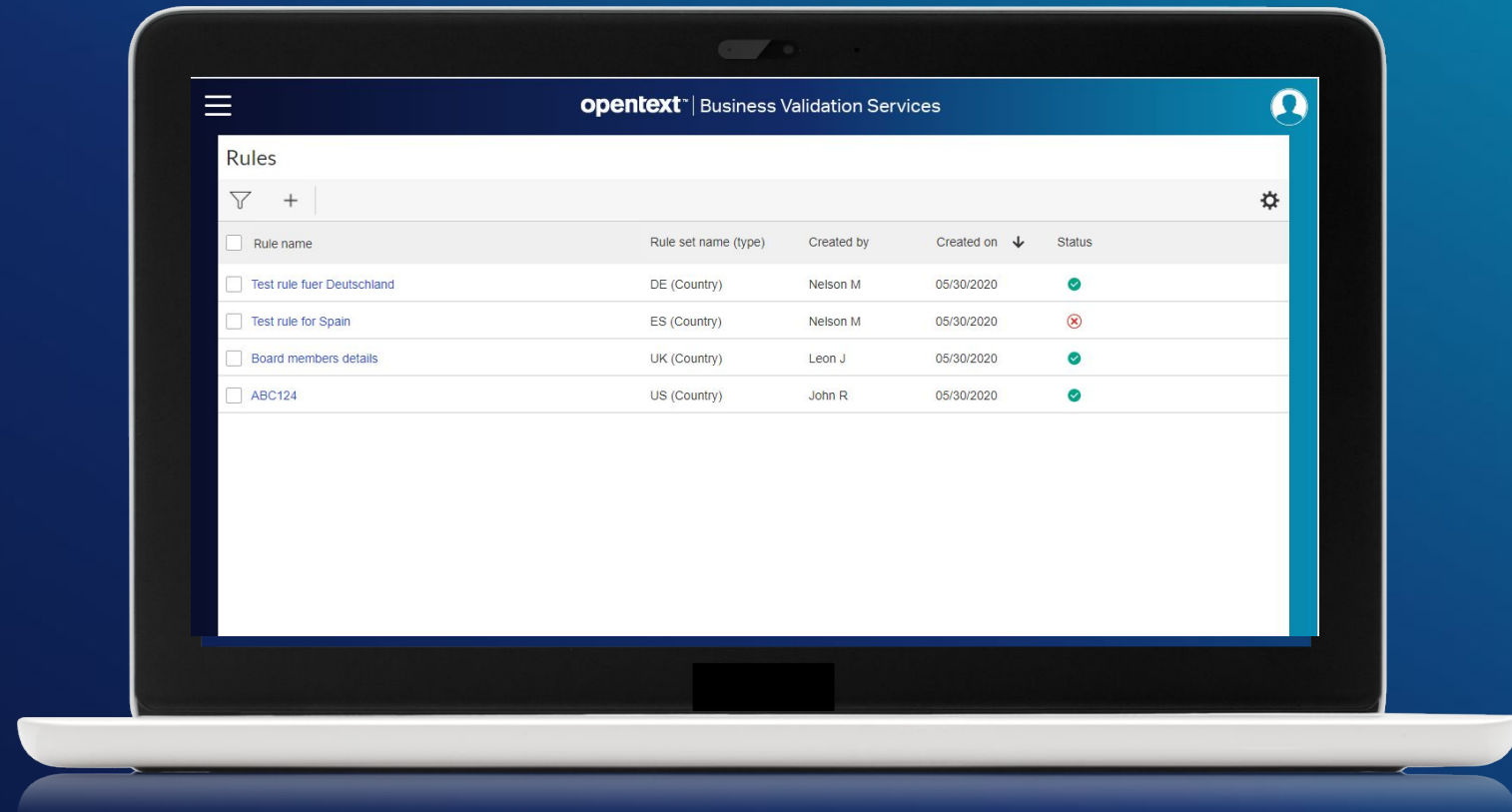
## Process

**Plan** to identify appropriate approaches and milestones for UCD into the overall design and development process. **Learn** and understand the user needs, expectations, context, technologies.

**Explore** and **Analyse** the user needs by building user profiles on gathered data, producing materials that will aid the outlining of the project IA, concept sketches, task flows, user journey, scenarios.

**Create** a solution based on the research findings, develop concept sketches, low fidelity prototypes, interactive prototypes, details design. **Refine** and evaluate design with stakeholders to obtain feedback. The **Usability Testing** technique uses to validate the solution across the user segment. **Deliver** complete design and produce deliverables.

# Showcase Business Validation Services



BVS is a rule engine originated with a requirement from BMW who requested for more than just mandatory validations to align with their compliance needs.

This resulted in BVS being developed in the year 2020, planned for pilot deployment by mid 2021.

- BVS helps other products to focus on their core product functionality and utilize BVS as the validation engine to be invoked as and when required.
- It removes dependency on engineering by making these validations configurable by authorized users (PS to start with and later extend to customers).
- Supports configuration of conditional validations in addition to the currently available mandatory, field presence checks.

The screenshots illustrate the opentext Business Validation Services interface. The top screenshot shows a 'Fields' table with the following data:

Field name	Data source type	Created on	Created by
<input type="checkbox"/> Lorem ipsum dolor sit amet	Input	05/30/2020	John R
<input type="checkbox"/> Finibus Bonorum et Malorum	Static	05/30/2020	John R

The middle screenshot shows the 'Create field' form with the following fields:

- \*Data source type: Static
- \*Field name: Lorem ipsum dolor sit amet
- \*Data type: String
- \*Value type: Key value pairs
- \*Value: 10, 0

The bottom screenshot shows the 'Audit' tab for the field 'Lorem ipsum dolor sit amet'. It displays a change log and a table of updated parameters.

Change log:

- 09/26/2021 V21.2.2 Current version
- Changed by Nelson M
- Reason for change: There is a request for change in field.

Updated parameter table:

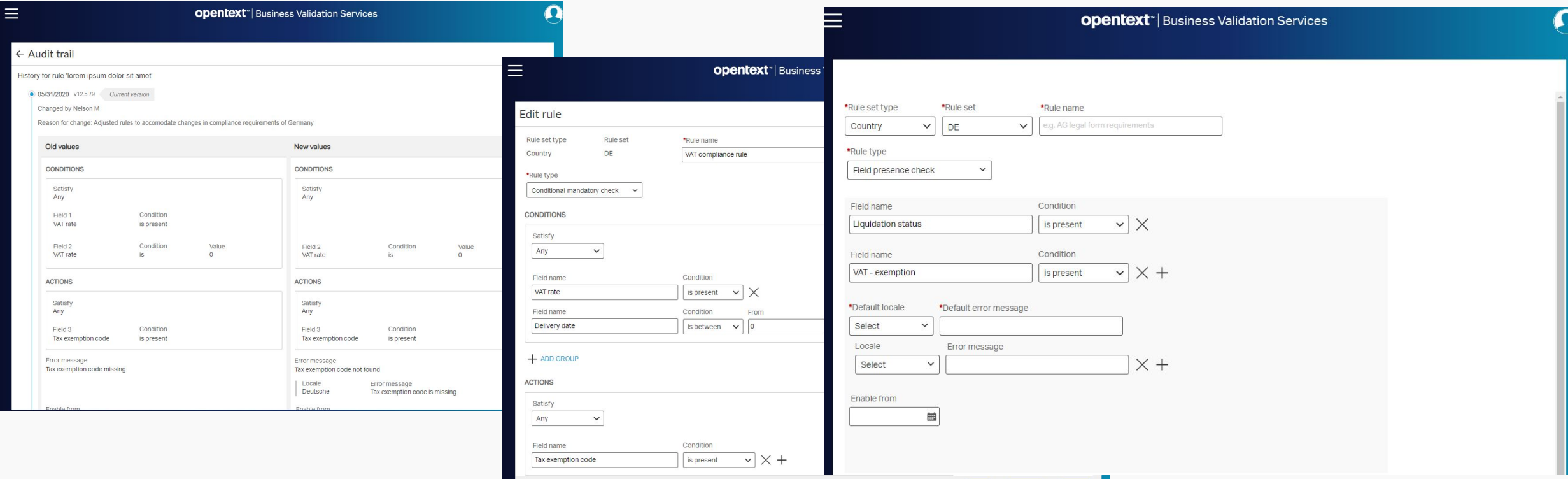
Updated parameter	Old value	New value
Field name	Lorem ipsum dolor sit amet	Perspicatis unde

Value type - key value pairs table:

Key	Value		Status
	Old	New	
AU	10, 0,	10, 0,15	Modified
AT	20, 10,	20, 10, 13, 0	Modified
BE	21, 6, 12, 0	-	Removed
BG	-	20, 9, 0	Added

Handover

BVS is a new product which is been developed for a year, where a few initial screens were developed with a mindset where users can utilise BVS validating the business documents. I noticed that the initial screens were having severe usability issues, which then became my primary focus to take into attention to the product manager and team with rationale and ask to go for incremental enhancements.





## Heuristic evaluation

Without any delay I started the heuristic evaluation on the developed screens and presented the analysis document in front of the product manager. Later included the cross functional team to get an understanding on what are those pitfalls and challenges, that needs immediate change. With all the valid analysis documents it was less tough than was expected, in context to push for the enhancements and a fresh JIRA was created for incremental enhancements.

opentext™ | Business Validation Services

\*Rule set type

Country

\*Rule set

DE

\*Rule name

e.g. AG legal form requirements

\*Rule type

Field presence check

Field name

Liquidation status

Condition

is present

×

Field name

VAT - exemption

Condition

is present

×

+

\*Default locale

Select

\*Default error message

Locale

Select

Error message

×

+

Enable from

Severity type - **Showstopper**

Problem Area - H1, H2, H4, H6, H7, H8

Problem description - The grouping fails to explain the visual hierarchy. Indentation is creating confusion by increasing intellectual load. Addition of extra rows will clutter the screen inducing the VIMM model. Too many actions without any clue is where user fails to perform within time.

Solution - Proper labelling and grouping to declutter the screen.

AS per OT the severity comes under **CRITICAL**

Problem description - For primary UI elements, there are obvious alignment issues, obvious inconsistencies with visual presentation; or, numerous minor cosmetic issues resulting in an unprofessional or sloppy appearance.

Gestal's principle of visual hierarchy missing.

Violation of Hick-Hyman's Law.

VIMM model induced.

### New screen suggested

Enable from

Check the field presence with paired condition

Field name Condition

Liquidationstatus is present +

Define error message

\*Default locale \*Default error message

Select

Other locale error message (optional) ☐

Other locale Error message

Select

×

Other locale Error message

Select

×

+



Heuristic evaluation

opentext™ | Business Validation Services

← Audit trail

Reason for change: Adjusted rules to accomodate changes in compliance requirements of Germany

Old values

CONDITIONS

Satisfy  
Any

Field 1  
VAT rate

Condition  
is present

Field 2  
VAT rate

Condition  
is

Value  
0

ACTIONS

Satisfy  
Any

Field 3  
Tax exemption code

Condition  
is present

Error message  
Tax exemption code missing

Enable from  
06/20/20

New values

CONDITIONS

Satisfy  
Any

Field 2  
VAT rate

Condition  
is

Value  
0

ACTIONS

Satisfy  
Any

Field 3  
Tax exemption code

Condition  
is present

Error message  
Tax exemption code not found

Locale  
Deutsche

Error message  
Tax exemption code is missing

Enable from  
06/20/20

Severity type - Showstopper

Problem Area - H2, H6, H8

Problem description - The grouping fails to explain the visual hierarchy. The repetitive information and gaps in between is confusing and creating maximum cognitive load to recall.

Not easy to identify with the paired values defined inside old and new.

Due to repetitive information there is information overload and a lot of eye scanning happens, increasing the visual load.

Solution - Proper grouping with minimal and sleek presentation.

VIMM model induced.

AS per OT the severity comes under **CRITICAL**

Problem description - For primary UI elements, there are obvious alignment issues, obvious inconsistencies with visual presentation; or, numerous minor cosmetic issues resulting in an unprofessional or sloppy appearance.

New screen suggested

opentext™ | Business Validation Services

← Lorem ipsum dolor sit amet ⏪ ⏩

View and Edit

Audit

10/26/2021 V21.2.2 Current version

Changed by Nelson M

Reason for change: There is a request for change in field.

Updated parameter	Old value	New value
Xpath	Lorem ipsum	Perspiciatis unde
Data type	String	Numeric

01/12/2021 V21.2.1

Created by John R

New ideas which is applauded

My contribution was recognised with a high-five award. Below are some of the new ideas that got high appreciation.

opentext™ | Business Validation Services

← Test ruler fuer Deutschland

View and EditAudit

Edit

\*Rule set

DE

\*Rule name

Test ruler fuer Deutsch

\*Rule type

Field value check

Test ruler fuer Deutschland

Audit

\*Rule name

Test ruler fuer Deutschland

Test ruler fuer Deutschland

Kinibus palorum

Ipsum dolor sit amet

Amet dolar

Sit ipsum dolor amet

Dolor sit amet

Perspiciatis unde

Lorem Ipsum

Bonorum et Malorum

Malorum Finibus

Load more 10

opentext™ | Business Validation Services

← Lorem ipsum dolor sit amet

View and EditAudit

10/26/2021 V21.2.2

Current version

Changed by Nelson M

Reason for change: There is a request for change in field.

Updated parameter	Old value	New value
Xpath	Lorem ipsum	Perspiciatis unde
Data type	String	Numeric

01/12/2021 V21.2.1

Created by John R

## The Beginning of a new requirement

I still remember the very first day when I got the JIRA notification mail and my product manager pinged me in the teams and wanted to discuss the JIRA created while setting a call between cross-functional teams.

That's how it started, first with 1:1 with product manager and later a collaborative discussion session including the product owner, the solution architect.

Being a listener on the very first day, I remember it was a confusing discussion on a few pieces of statements that came from the client-side (BMW). The users are Opentext professional services.

It is then announced that BVS is going to play a major role in future when it will become customer facing and most of the Opentext products can use BVS for the validations purposes. Hence I had a clear idea that the user experience need to focus on maximum flexibility and easy to use with clear upfront guided navigation.

### Use Case:

1. Check if a value satisfies minimum or maximum length,
2. Verify format of date, date time, SSN, Zip code etc.
3. Verify that the field begins, contains, or ends with a specific string.

### EXAMPLES

- Invoice number should start with 'INV' + 8 digits.
- Invoice should be between 6 – 12 digits
- Exempt reason can be any of

- Exempt intra-Community supply of goods according to article 138 Directive 2006/112/EC
- Supply of goods under the simplified triangulation regime according to article 141 Directive 2006/112/EC
- VAT due by the contracting party according to article 197 Directive 2006/112/EC
- Export supply of goods exempt from VAT according to article 146 Directive 2006/112/EC
- Triangulation

In addition, the reason in the EDI file may not contain the whole sentence.

- Eg: The exemption reason implying "Exempt ~~intra-Community~~ supply of goods according to article 138 Directive 2006/112/EC" maybe sent in the EDI file in any of the following ways or more.
- Exempt ~~intra-Community~~
- Exempt ~~intra-Community~~ supply of goods
- Exempt ~~intra-Community~~ supply of goods according to article 138
- Exempt ~~intra-Community~~ supply of goods according to article 138 Directive 2006/112/EC

## Approach & Strategy

Within a few discussions, I concluded that the first task would be to break the ice by cracking the requirement statements first.

Hence, I planned to document things from the first day on the clues and the point of direction that will come out during these discussion sessions. Going forward that can help me to refer, identify, eliminate accordingly in the critical scenarios.

## My Role

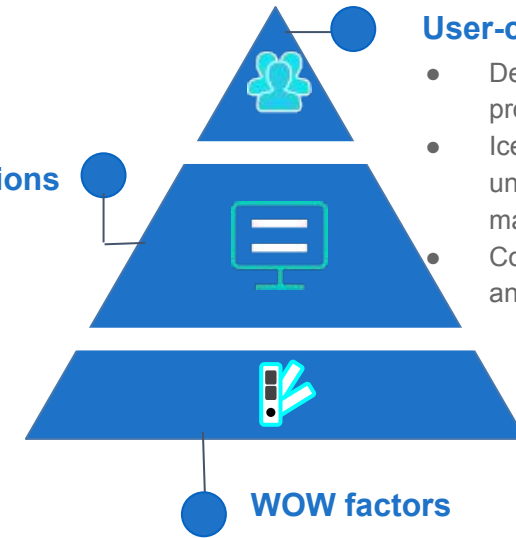
As an individual contributor, I report and collaborated with the product manager, product owner, solution architect. As part of this work, I facilitated collaborative sessions & usability sessions and documented them.

Most of the validations of my ideas were through 1:1 meetings, design critique sessions, and team presentations.

## Design maturity

### Interface design and interactions

- Convert research input to concept sketches, journeys, interactive prototypes.
- Design solution validation/ usability testing of a specific section or flow.



### User-centric

- Design is seen as a way to solve problems.
- Ice breaking sessions to understand the business need mapping with user needs.
- Considering user's mental model and built to that.

### WOW factors

- Presentation is key for success.
- Visual colours, images, fonts should reflect the branding.
- Design system first.

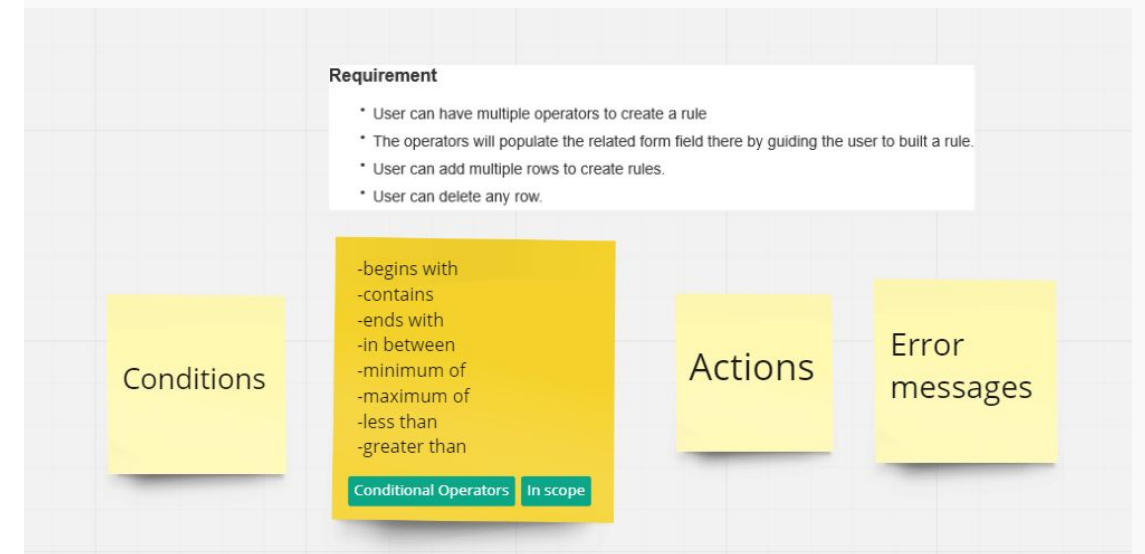
## Ice Breaking Session

The ice-breaking session was based on the requirement statement where it was mentioned about the operators which can be paired with multiple fields and values to create a rule.

In the discussion probe in and around the statement and ask to share their views. We discussed the probable look around to connect the dots of every point discussed in the session.

The sessions came out fruitful. I could able to understand and gather different views and examples shared by the team and can relate them to the existing requirement.

We discussed the scopes and try to understand what are the elements that are needed to create a conditional rule. Tried to identify the new elements while keeping the existing flow, also discussed what could be the probable attributes for these new elements.



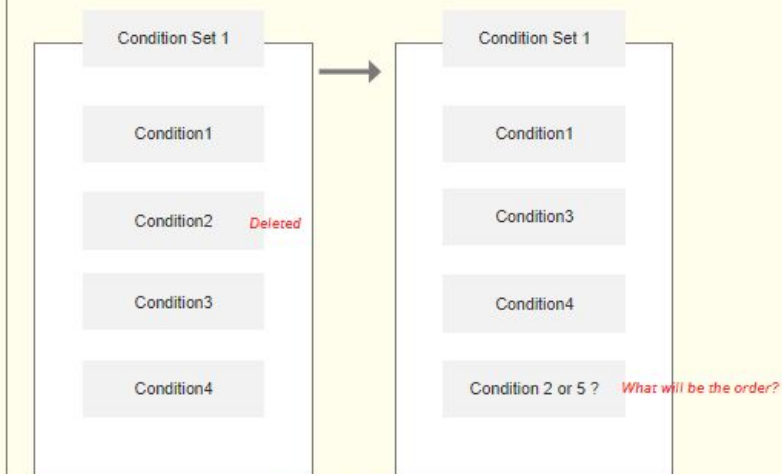
### Clarity needed on the business logic for the content and order.

User creates 3 conditions under a Condition set.

What would be the suitable naming and what should be the order?

If consider the label is condition and order comes in chronological then it will be Condition1.

Scenario - If condition set has 4 conditions as cond1, cond2, cond3 and cond4. Suppose cond2 is deleted among them. What will be the order? [refer below concept sketches.](#)



[Concept sketches.](#)

## Connecting the dots

### Research

1. **Field name** - The first part allow user to describe what he/she wants to achieve with this rule using the conditional logic.

2. **Condition** - Is the statement refer to a field and must be fulfilled. Condition consists of Operators.

**Operator** - different operators are used to evaluate the condition.

3. **Value type and Value** - The way Value type and value will be defined are going to vary according to the type of field and the type of operator.

Mapping the Conditions/Operators with their related data fields (probable) -

Conditions/Operators	Definition	Value type
Equals / Does not equal	Checks if the answer is an exact match to another value.	<ul style="list-style-type: none"> <li>Freetext</li> <li>Integer/Decimal/Currency</li> <li>Date/Time/Date Time</li> <li>Duration</li> </ul>
Matches / Does not match		
On		
Before		
After		

Field name Condition Value type Value

Field name Condition Value type Value type

Field name Condition Value type Value type

Conditions/Operators	Definition	Value type
Less than	Checks if the answer is lower than another value / higher than another value / falls between (but is not equal to) two other values.	<ul style="list-style-type: none"> <li>Integer/Decimal/Currency</li> <li>Date/Time/Date Time</li> <li>Duration</li> </ul>
Greater than		
Between		

Field name Condition Value type Value

Field name Condition Value type Data type Value

Field name Condition Value type Value

Conditions/Operators	Definition	Value type
Begins with	Checks if the answer begins with / ends with of the given value.	<ul style="list-style-type: none"> <li>Freetext</li> <li>Integer/Decimal/Currency</li> <li>Email / Phone number</li> </ul>
Ends with		

Field name	Condition	Value type	Value
<input type="text" value="Lorem ipsum"/>	<input type="text" value="Begins with"/>	<input type="text" value="Free text"/>	<input type="text" value="INV"/>

Field name	Condition	Value type	Value type
<input type="text" value="Lorem ipsum"/>	<input type="text" value="Ends with"/>	<input type="text" value="Email / Phone number"/>	<input type="text" value="p.s@g.com"/>

Conditions/Operators	Definition	Value type
Minimum of	Checks if the answer is minimum of / maximum of the given value.	<ul style="list-style-type: none"> <li>Free text</li> <li>Integer/Decimal/Currency</li> </ul>
Maximum of		

Field name	Condition	Value type	Value
<input type="text" value="Lorem ipsum"/>	<input type="text" value="Minimum of"/>	<input type="text" value="Integer/Decimal/Currency"/>	<input type="text" value="\$15"/>

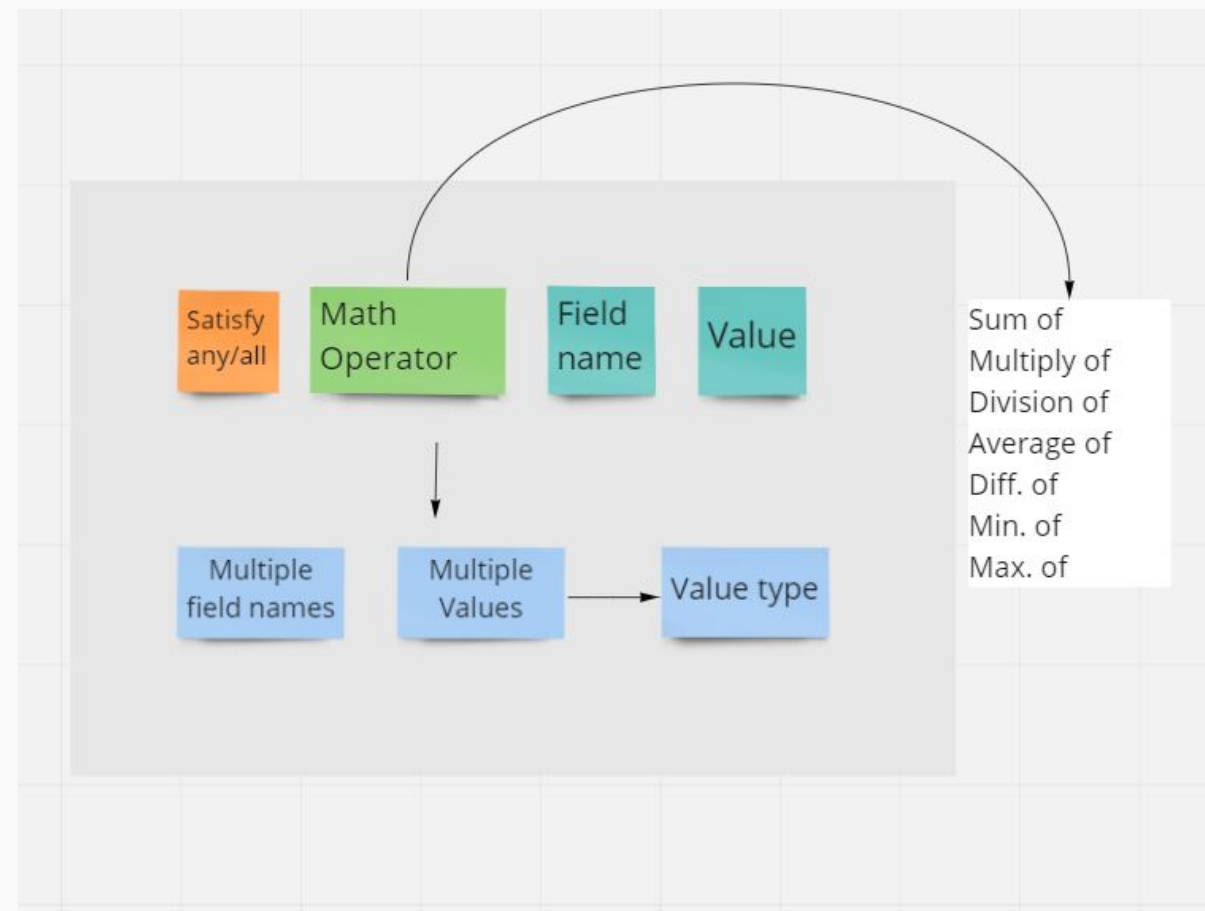
Field name	Condition	Value type	Value
<input type="text" value="Lorem ipsum"/>	<input type="text" value="Maximum of"/>	<input type="text" value="Free text"/>	<input type="text" value="8 digits"/>



## Brainstorming - redefining the use case

While explaining the research about the different conditional operators that support the field type or value type, and giving examples of how it might work with conditional statements, the cross-functional team could identify that the conditional operators are not only the best combinations to create the conditional rule.

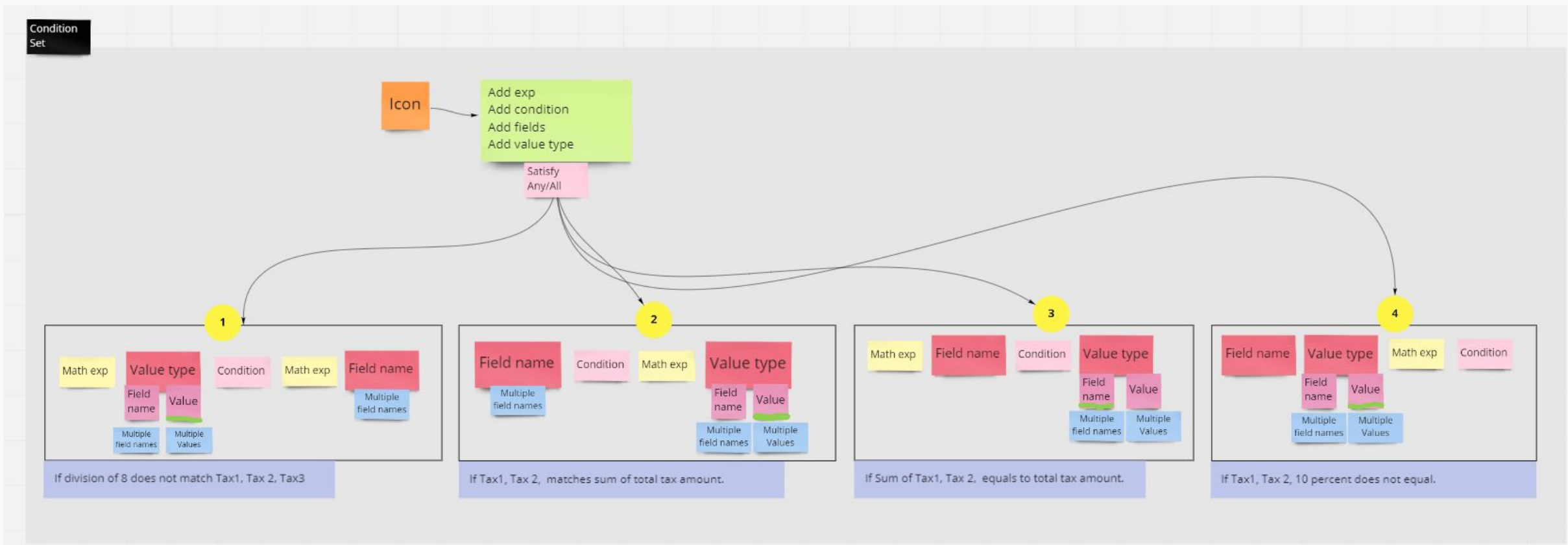
To create a conditional statement on country-specific TAX rules or VAT rules we need to include math operators. As there was more clarity on the requirement side, we gradually started exploring the math operators and the probable use cases.





## Brainstorming - redefining the use case

We started discussing the other attributes that can pair with the math operators, combinations that can be needed from the user side to create a conditional rule. Moving forward, with these existing data the Product Manager was able to gather some more insights from the business side and we then gradually started working on those points.

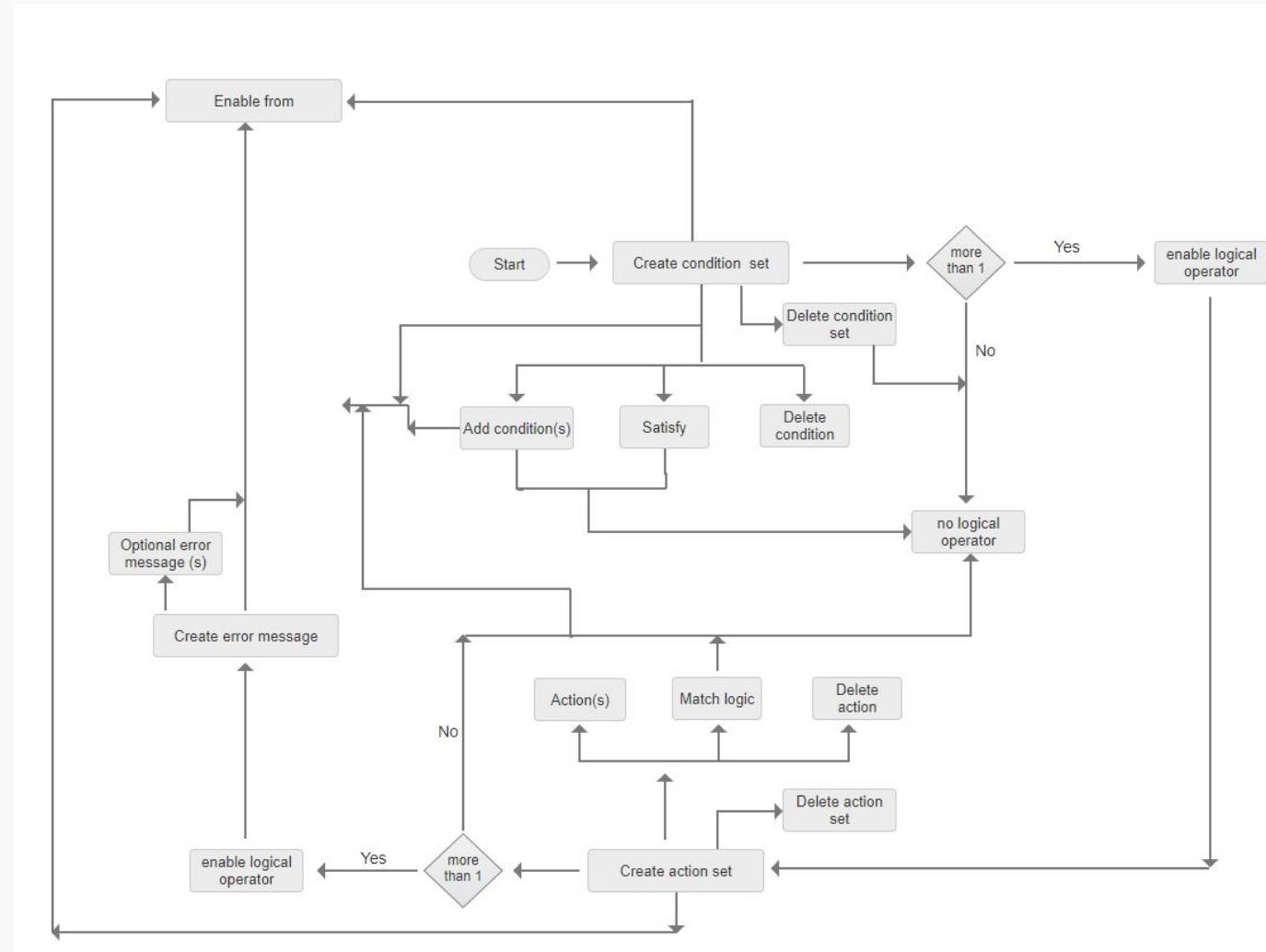


## User touch points - task flow

Empathy is the core of understanding - the right problems to the right audience.

On that note, I created the task flow to identify the traces of user's possible paths through sequences of tasks (from creating multiple conditions in a condition set and then defining the actions within the action set) and decision points (on defining the error messages) while accomplishing their goal.

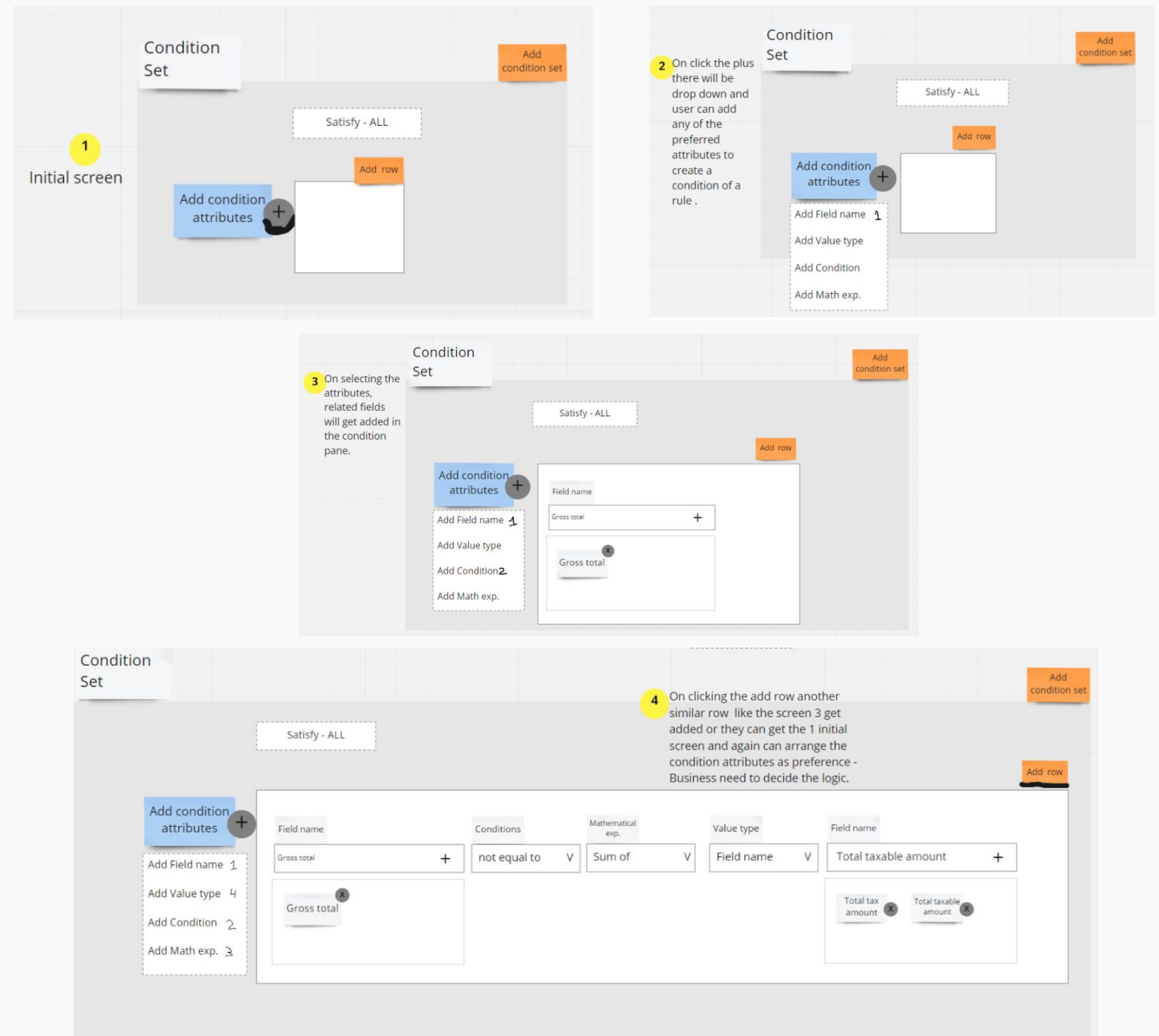
It validates the understanding of the user goals from one touchpoint to another and helps identify the common scenarios.



## Concept sketches

The best way I have found to quickly, easily, and comprehensively communicate with my team: share rough concept sketches, which is quick and time-saving.

Not only does this help me articulate how I plan to tackle certain scenarios, but thinking out loud on paper, allows everyone—myself, the cross-functional team to see how ideas begin, evolve, and finally crystalize into the solution we've been working on.



## Concept sketches

Rough sketches were created while discovering new ideas, I could use my imagination at the most to put my creativity to action.

Later when I introduced all of these sketches, collaboratively we discussed the feasibility of the framework, development time and effort etc. involved and can identify the best out of rest.

### CONDITIONS

Sketch 1 shows the initial interface. At the top, there is a 'Satisfy' label and a dropdown menu with 'Select' and a downward arrow. Below this is a dashed rectangular box. To the left of the box is a grey circle with an 'X' and a purple hand cursor pointing at it. To the right of the box is the text 'Add row'.

### CONDITIONS

Sketch 2 shows the interface with a menu overlay. The menu is a dark grey rectangle with white text, listing: 'Add field name', 'Add Value type', 'Add Conditions', and 'Add Mathematical expression'. A purple hand cursor is pointing at the 'Add Value type' option. The rest of the interface is the same as in Sketch 1.

### CONDITIONS

Sketch 3 shows the interface with a field name input. The dashed box now contains a 'Field name' label, a text input field with a '+' button on the right, and a grey rectangle with an 'X' in the bottom right corner. A purple hand cursor is pointing at the 'X' icon on the left. The 'Add row' text is still present.

### CONDITIONS

Sketch 4 shows the interface with a condition dropdown. The dashed box now contains a 'Field name' input and a 'Condition' dropdown menu with 'Select' and a downward arrow. A purple hand cursor is pointing at the 'X' icon on the right. The 'Add row' text is still present.

### CONDITIONS

Sketch 5 shows the interface with a menu overlay. The menu is a dark grey rectangle with white text, listing: 'Add fields', 'Add Value type', 'Add Conditions', and 'Add Mathematical expression'. A purple hand cursor is pointing at the 'Add Value type' option. The dashed box contains a 'name' label, a text input field with a '+' button, and a grey rectangle with an 'X' in the bottom right corner. The 'Add row' text is still present.

## Critical thinking and concept sketches

As most of the elements are fixed and now we have a fair idea of how the rule builder may look like catering all the possible requirements that is there from the user side.

Now further discussion was needed on the functional requirements on deletion.

The order get changed when there is deletion of any row from a set of condition.

Created the concept screens to give a clear picture on how it might affect the order also, to initiate a discussion on the feasibility.

It also helped to get the attention of the stakeholders and cross functional team on what could be the possible experiences from user's perspective.

### Conditional pane Critical thinking on Functional Requirements

#### Visual Hierarchy

- If user needs to add another condition inside the same set from edit then what?  
User can add a new condition, will appear as incremental order.  
So if 1-2-3 exists then addition of another condition, condition 4 will get added to the existing order.
- What if between 1-2-3 conditions of a condition set, the condition 2 is deleted... and need to add another condition to that set?  
The order becomes 1-3 after deletion of row2.  
If we keep the same row order which is deleted then it will appear like 1-3-2?  
User get the clue with such row order, but it is a lot of stress when many rows involved?  
A lot of cognitive stress with jumbled up row number!
- What if when condition row 2 is deleted, ar  
If the order will be same again 1-2-3 and r  
The next addition of a row order can be co  
User gets a clear understanding of the ordi  
information on the deleted attributes.
- How it might appear inside edit screen- the  
All rows will have the editable fields except  
It helps user to focus more when many cor  
Less stress and more easy to connect the

#### • Visual hierarchy logic in the edit screen

Condition set Add condition set

Condition1	
<input type="checkbox"/> Field name	<input type="checkbox"/> Conditions
<input type="checkbox"/> Value type	<input type="checkbox"/> Value

X deleted

Condition2	
<input type="checkbox"/> Field name	<input type="checkbox"/> Conditions
<input type="checkbox"/> Value type	<input type="checkbox"/> Value

X

Condition3	
<input type="checkbox"/> Field name	<input type="checkbox"/> Conditions
<input type="checkbox"/> Value type	<input type="checkbox"/> Value

X +

Condition set Add condition set

Condition2	
<input type="checkbox"/> Field name	<input type="checkbox"/> Conditions
<input type="checkbox"/> Value type	<input type="checkbox"/> Value

X

Condition3	
<input type="checkbox"/> Field name	<input type="checkbox"/> Conditions
<input type="checkbox"/> Value type	<input type="checkbox"/> Value

X

Condition1	
<input type="checkbox"/> Field name	<input type="checkbox"/> Conditions
<input type="checkbox"/> Value type	<input type="checkbox"/> Value

X +

Condition set Add condition set

Condition1	
<input type="checkbox"/> Field name	<input type="checkbox"/> Conditions
<input type="checkbox"/> Value type	<input type="checkbox"/> Value

X Deleted

Condition2	
<input type="checkbox"/> Field name	<input type="checkbox"/> Conditions
<input type="checkbox"/> Value type	<input type="checkbox"/> Value

X

Condition3	
<input type="checkbox"/> Field name	<input type="checkbox"/> Conditions
<input type="checkbox"/> Value type	<input type="checkbox"/> Value

X

Condition4	
<input type="checkbox"/> Field name	<input type="checkbox"/> Conditions
<input type="checkbox"/> Value type	<input type="checkbox"/> Value

X + added

Condition set Add condition set

Condition2	
<input type="checkbox"/> Field name	<input type="checkbox"/> Conditions
<input type="checkbox"/> Value type	<input type="checkbox"/> Value

X

Condition3	
<input type="checkbox"/> Field name	<input type="checkbox"/> Conditions
<input type="checkbox"/> Value type	<input type="checkbox"/> Value

X

Condition4	
<input type="checkbox"/> Field name	<input type="checkbox"/> Conditions
<input type="checkbox"/> Value type	<input type="checkbox"/> Value

X +

## Critical thinking on risk factors

This is prepared utilising the critical thinking process where I have created long list of questions of a particular “Deletion” state.

Where I have drill down into the row deletion and the limitations related to that.

These are vital points which needs attention from the stakeholder and cross functional teams to discuss. These points needs business approvals as to determine what are the restrictions to cater a rue builder.

Still it is in discussion from business side and the work is on little slower as the rule builder must have a business justification on the limitations of row deletion to proceed for further steps.

### Mitigating risk

- What if, user want to delete all 4rows of condition in condition set1? (Suppose there are 3 such sets in the condition pane.)  
User can delete 4 rows and again start creating inside set1.
  1. Why user delete all the conditions when he can edit each one? - Restrict delete, push for edit.
  2. Why user delete 1condition among 4? - business requirement / by mistake.
  3. Is there any undo action there if user want to retain the deleted rule if it is by mistake? - No
  3. If alert while deleting is preventive or annoying? - It will be preventive as the deletion impact by mistake is huge and hence the alert will help users recognize, diagnose, and recover from errors, other side it will push the user not to delete rather edit.
- What if all the 4 or more rows of condition which needs edit and user find it difficult to erase all the editable fields data first and then fill with new. So to overcome this situation user decide to delete all conditions inside the set?  
In that case user may find the alerts preventive, as user can see the suggestion inside alert for deleting the condition set incase he needs to delete all the rows.  
Hence user is guided well to delete the set and create a new condition set with new rows of condition.
- What if there is only one set, are we allowing user to delete?  
When only one set of condition and user allowed to delete, then there be no initiator to create a set again.  
When there is more than one condition set user have flexibility to delete any one of the sets.  
If there is one set with rows of condition and he needs to delete the set then he will click on "add condition set" to initiate another set and then can delete the previous.
- How would user know that he needs to add another set to delete the existing set?  
In alert if we can accomodate the detail deletion content.  
Alternative, these informations can be present as "infotip".
- If the condition-action-errormsg is interdependent?  
Yes, user have to define the conditions first and based on that action and error message can be defined.  
Incase there are many condition sets defined, rows of conditions within it, then user need to scroll down to get to the next Action pane.
- What if user created 8 sets of condition and now wants to define the actions and the long scroll is bothering him?  
User can collapse the Define condition pane, to eliminate the long scroll.  
User can focus more into one pane when many sets are defined inside.



## Ideate and Design

### Flexible and Stay Focused

With a better approach to building the rule logic by accessing the information right on the rules view without having to jump into other tabs constantly.

Users can have combinations of rule attributes which makes the user's task easy and flexible to create a rule that is based on multiple condition sets paring with different conditions defining them with logical operators and actions. If these conditional statements fail, the user can set the error message that can only get executed on failed circumstances.

The screenshot displays the 'Create rule' interface within the opentext Business Validation Services application. The interface is structured as follows:

- Header:** The top bar features the opentext logo and 'Business Validation Services' text on the left, and a user profile icon on the right.
- Form Fields:**
  - Rule set:** A dropdown menu currently showing 'DE'.
  - Rule name:** An empty text input field.
  - Rule type:** A dropdown menu currently showing 'Conditional check'.
  - Enable from:** A date picker icon.
- Define Conditions Section:**
  - Condition Set 1:** A section header with an 'Add condition set' link to its right.
  - Satisfy:** Two buttons, 'All' (selected) and 'Any'.
  - Condition1:** A container for individual conditions with a '+' icon to add more.

Field name (LHS)	Condition	Value type	Value
<input type="checkbox"/> <input type="text"/>	Select ▼	<input checked="" type="checkbox"/> Value ▼	<input type="text"/>

- Footer:** Two buttons, 'Create' (highlighted in blue) and 'Cancel'.



## Ideate and Design

As **space matters** and to make the screen less cluttered, the solution is designed very sleek and lightweight yet with multiple options to play around in a delightful way.

The screenshot shows the 'Create rule' interface in the opentext Business Validation Services application. The interface is designed to be sleek and lightweight. It features a header with the opentext logo and the text 'Business Validation Services'. The main content area is titled 'Create rule' and contains two condition sets. The first condition set, 'Condition Set 1', has a 'Satisfy' dropdown set to 'All' and a 'Delete Set' link. Below this, there is a 'Condition1' section with a table-like structure for defining conditions. The table has columns for 'Expressions (LHS)', 'Field name / field name & value', 'Condition', 'Expressions (RHS)', and 'Field name and/or value'. The first row shows a 'Select' dropdown, a text input field, a 'Select' dropdown, another 'Select' dropdown, and a text input field. Below the table, there are 'Field1' and 'Value1' labels with 'x' icons. The second condition set, 'Condition Set 2', also has a 'Satisfy' dropdown set to 'All' and an 'Add condition set' link. It contains a 'Condition1' section with a table for defining conditions. The table has columns for 'Field name (LHS)', 'Condition', 'Value type', and 'Value'. The first row shows a text input field, a 'Select' dropdown, a 'Value' dropdown, and a text input field. At the bottom of the interface, there are 'Create' and 'Cancel' buttons.

The screenshot shows the 'Create rule' interface in the opentext Business Validation Services application, focusing on the 'Define Actions' and 'Define Error Message' sections. The interface is designed to be sleek and lightweight. It features a header with the opentext logo and the text 'Business Validation Services'. The main content area is titled 'Create rule' and contains two sections. The first section, 'Define Actions', has an 'Action Set 1' header with an 'Add Action set' link. Below this, there is an 'Action1' section with a table-like structure for defining actions. The table has columns for 'Field name (LHS)', 'Condition', 'Value type', and 'Value'. The first row shows a text input field, a 'Select' dropdown, a 'Value' dropdown, and a text input field. The second section, 'Define Error Message', has a 'Default locale' dropdown set to 'Select' and a 'Default error message' text input field. Below this, there is an 'Other locale' dropdown set to 'Select' and an 'Error message' text input field. At the bottom of the interface, there are 'Create' and 'Cancel' buttons.

## Usability test

So, once a solution is ready to get into the development phase, I wanted to validate the solution from the real users. My PM was happy and helped me on getting the real user's details. We conducted the usability test with 3 real users where the users were asked to create a conditional rule with different use cases provided by my PM, and I was observing where the user might encounter problems or experience confusion.

When they finished their tasks, I asked a few questions to understand if anything that is missed or needed to be accommodated or enhanced in the design. They were able to finish the task with no help and was happy with the provided solution.

UT questions and feedbacks -

Do you think the solution has captured all your task related inputs, if not can you please share what is your expectation? **Yes, this is good, I could get a lot of combinations for the field names and values, which is really a cool feature.**

Is there anything that you find need improvement, or any suggestion that you want to provide? **Nothing as such, looks good to me.**

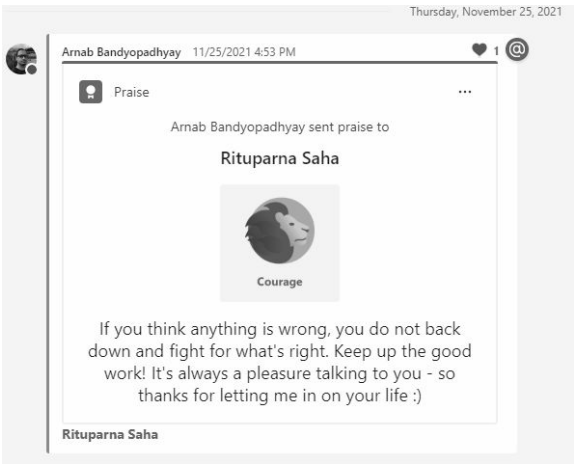
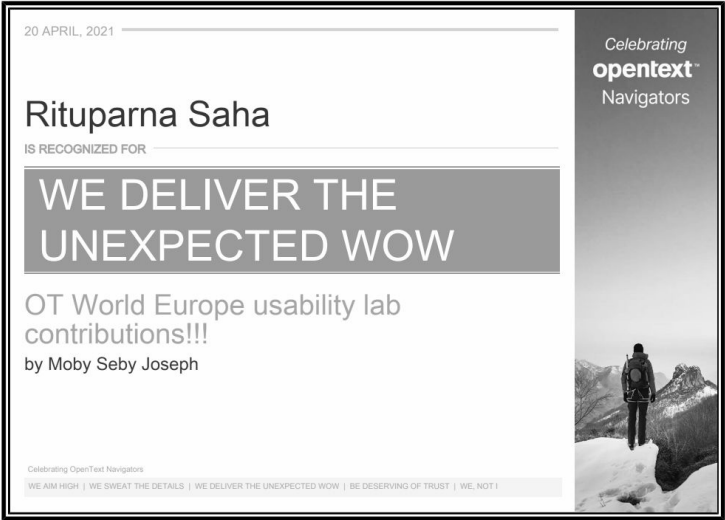
Is there any more mathematical expression that need to be added? **There is already a lot of options which I do not even know if that is needed or not, for now I can say few are useful, and others might be helpful in certain scenarios.**

How satisfied are you with the available solution? **It looks great.**

Do you think the LHS and RHS may confuse a user? - **No, not at all, we know all these terms.**

Any feedback on the tooltip for LHS and RHS to help a new user? - **No, I do not think it is needed, anyone who is performing these actions might be aware of these terms.**

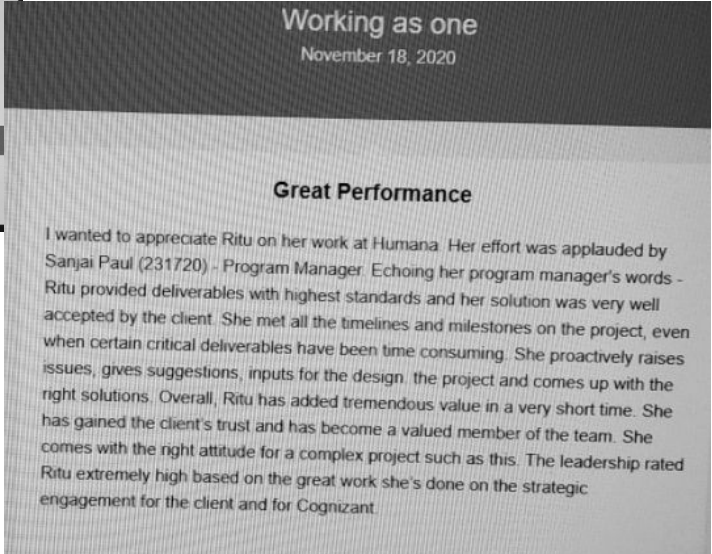
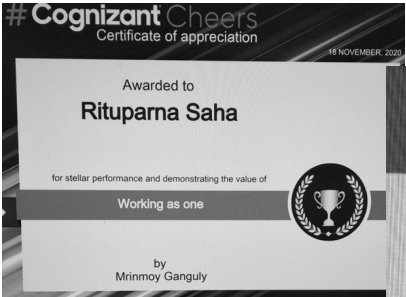
Some of the Recognitions that ignites my passion towards problem solving



"I did want to take a moment and say thank you to the team. You all did a fantastic job with Sprint 15!! The stakeholder demo went very well and I think everyone was impressed with the amount of progress that has been made so far. I am thrilled with the progress!

Keep up the great work! This project is delivering such value to our company and is a key initiative to help transform how we support our brand partners."

Karen (Lowe) Ashbrook  
Director, Digital Capabilities | Marketing Products & Capabilities  
Alliance Data Card Services



"Ritu- This is great work"  
Tanveer, Saad  
POI Provision, RET

"Very impressed with the UX side of the team and feel that we're all working together in a unified way."  
Lechitsky, Allen  
Director - CRM, EAS Salesforce PMG

"Condering these efforts and going forward in the implementation phase."  
Mochi, Venugopal  
Project Manager, EAS Salesforce

"Agreed.. Kudos to the UX team"  
Bose, Indraneel  
Engagement Lead, Cognizant Infra Services

# Thank You

Mobile : **9433549744**

Mail : [discover.footprints@gmail.com](mailto:discover.footprints@gmail.com)