IDS Maker

Web solution

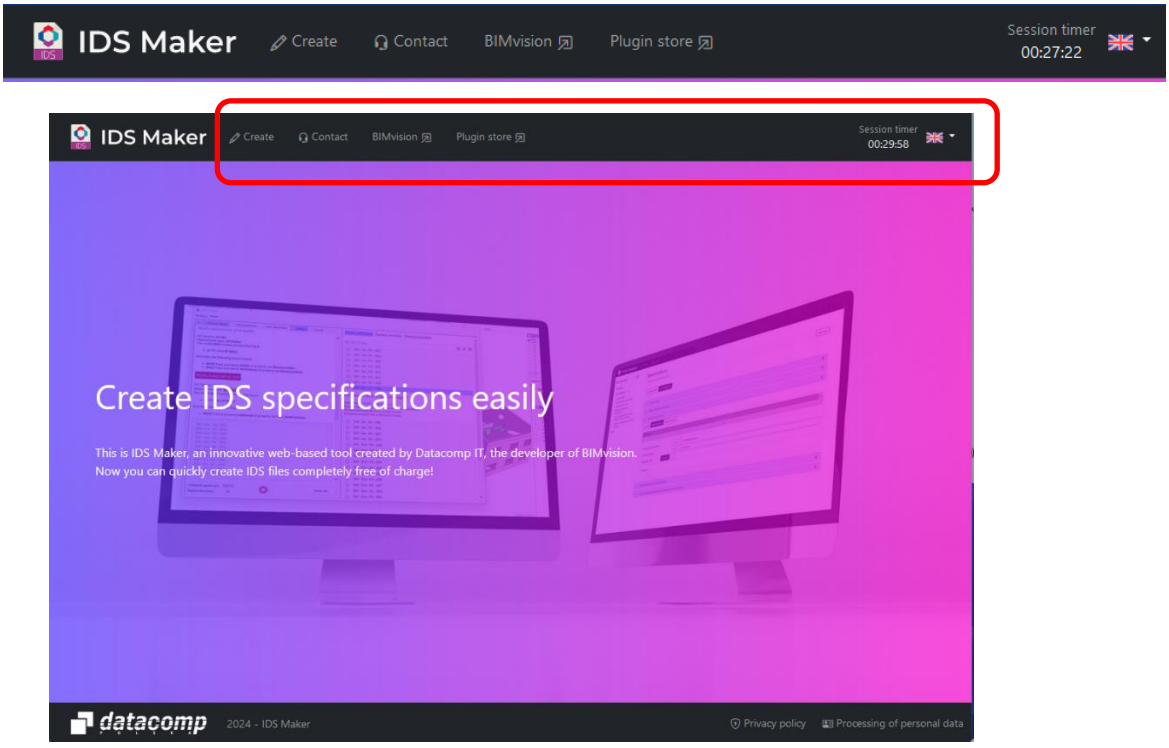
The IDS Maker plugin allows you to create an IDS requirements file. Information Delivery Specification is a special format developed by buildingSMART that enables you to define the conditions specific groups of elements in the IFC model must meet.

Such an IDS requirements file created in IDS Maker can be further used to check the correctness of the IFC model, e.g. using the BIMvision IDS Checker plugin.

User interface

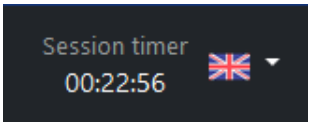
After going to the website: ids.bimvision.eu the user will see a welcome page. In the top bar there is a menu. There are buttons such as:

- **Create** – go to the main part of the site, **IDS file creator**
- Contact – contact information
- BIMvision – button linking to the BIMvision home page
- Plugin store – redirection to Plugin store
- <Flag icon> – change interface language



The program interface also includes a **Session timer** in the page's upper right corner.

This is an important element indicating the time of "no activity" on the page. The IDS Maker page in the non-logging version allows the creation of an IDS file continuously, without significant interruptions in work. The session will automatically expire after 20 minutes without any activity on the page.



After pressing the "Create" button, the user will see the **main interface.** In addition to the top menu bar, there is also a view of the IDS file structure on the left side.

The IDS structure consists of:

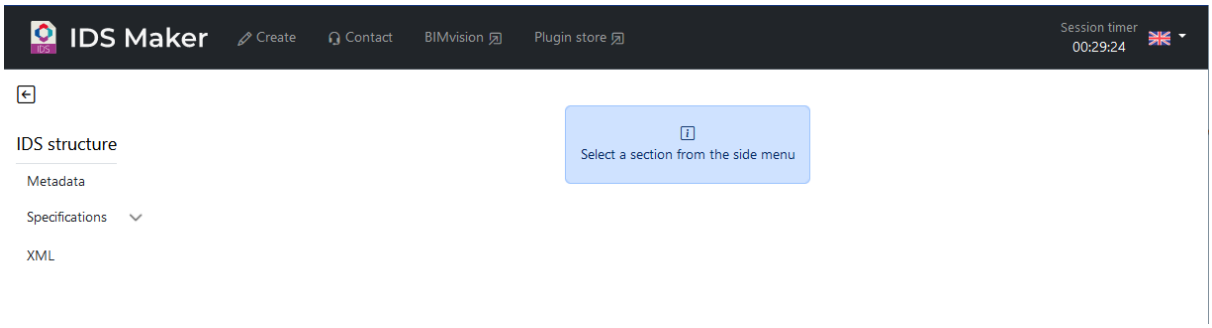
- Metadata – general data regarding the entire IDS file being created
- Specifications – a part that allows you to create and view specifications (as you add specifications, they will appear as subsequent items in this section)
- XML – access to the record of the IDS file being created in XML form and the ability to download it

Creating an IDS file

Creating an IDS file involves completing the data in the Metadata section and creating individual specifications.

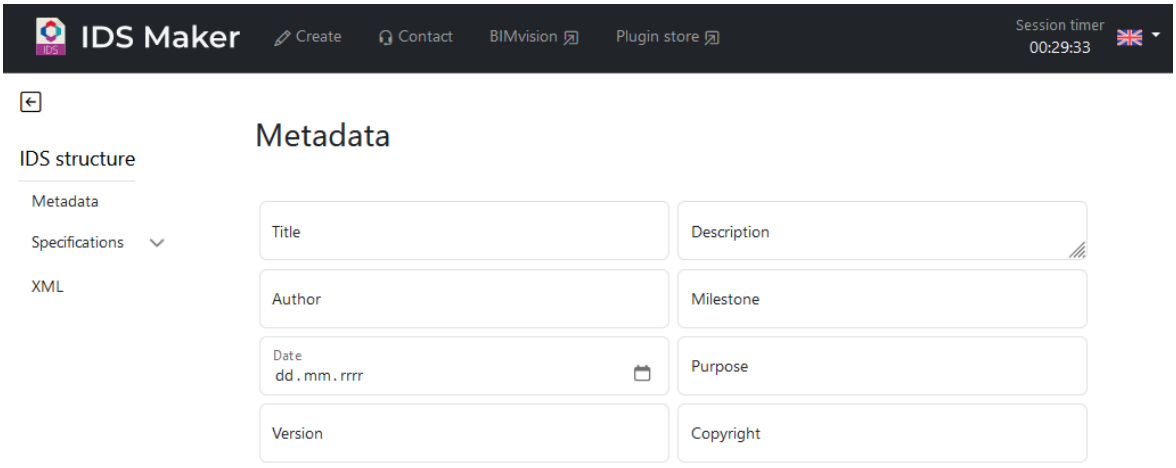
Specifications are components of an IDS file that contain information about the conditions that specific groups of elements must meet. Their structure has a strict logic and allows for defining requirements for the information content of elements in the IFC model in various ways.

The final step in creating an IDS file is to save it as an external file in the form of XML.

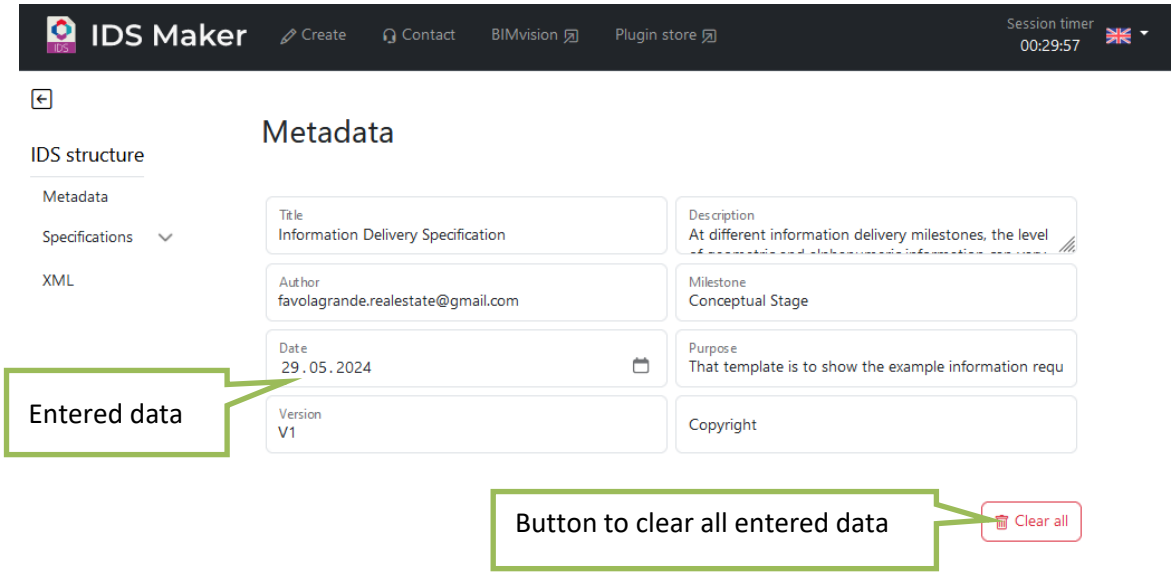


Creating an IDS file – Metadata

In the Metadata section, the user enters information about the IDS file being created. This includes information such as the title, the author of the IDS file (in the form of an email address), the creation date, the IDS file version number, and additional information such as a description, milestone, purpose, and copyright.



Once some data has been entered, an additional button will appear in the interface to quickly clear it.

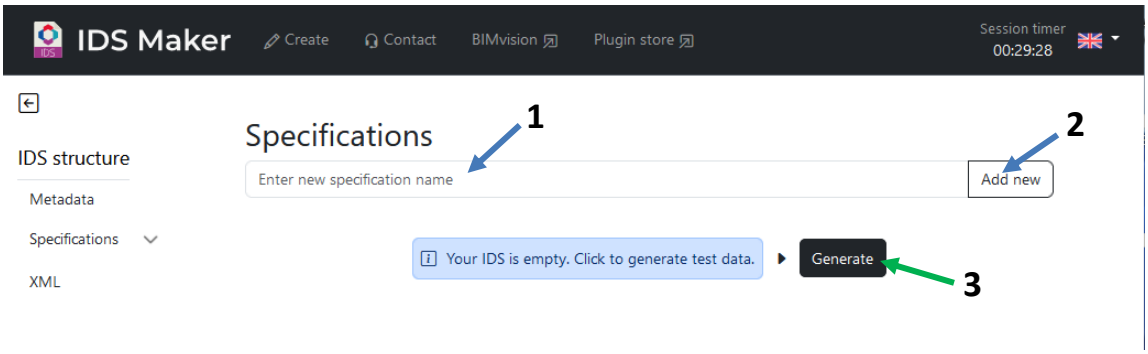


Creating an IDS file – Specifications

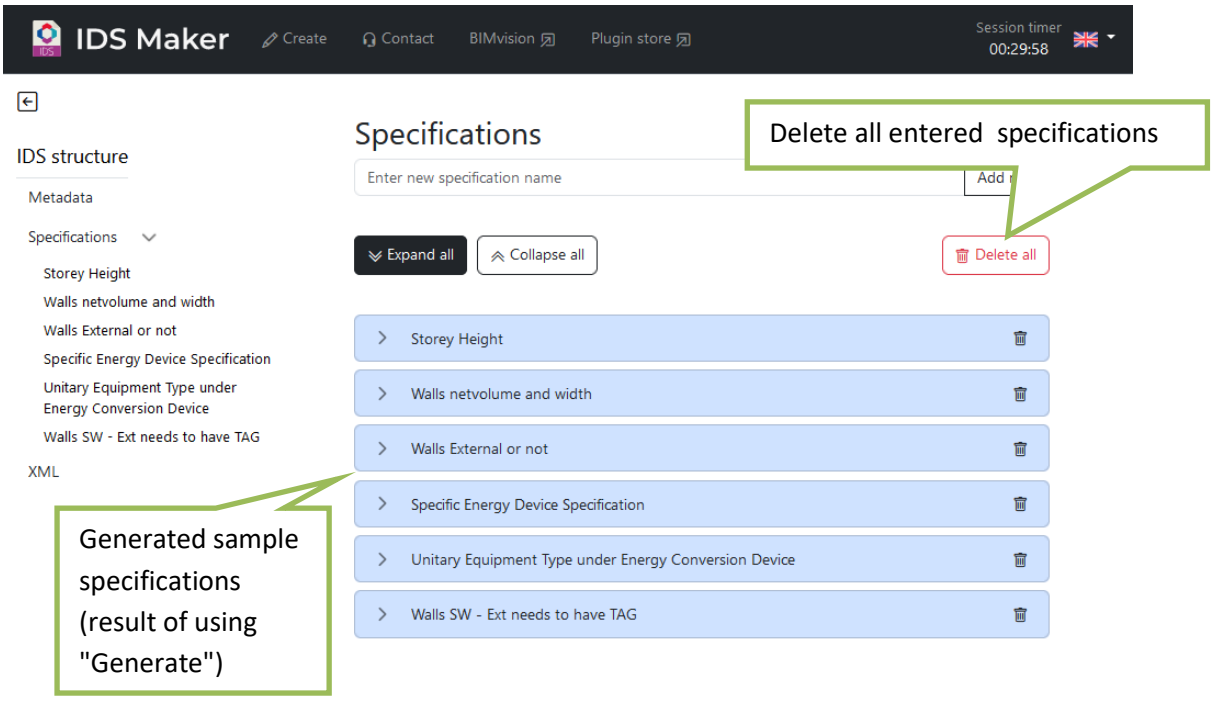
Once in the Specifications section, the user can begin creating individual new specifications.

To add the first new specification, enter its name in the "Enter new specification name"(1) field and click "Add new"(2). At this point, the first specification will appear, which you can work on, in order to enter the appropriate information, dependencies and conditions.

If you would like to first see what different specifications might look like, you can select the "Generate" button (3). This will generate several different sample specifications that you can review or modify, for example.

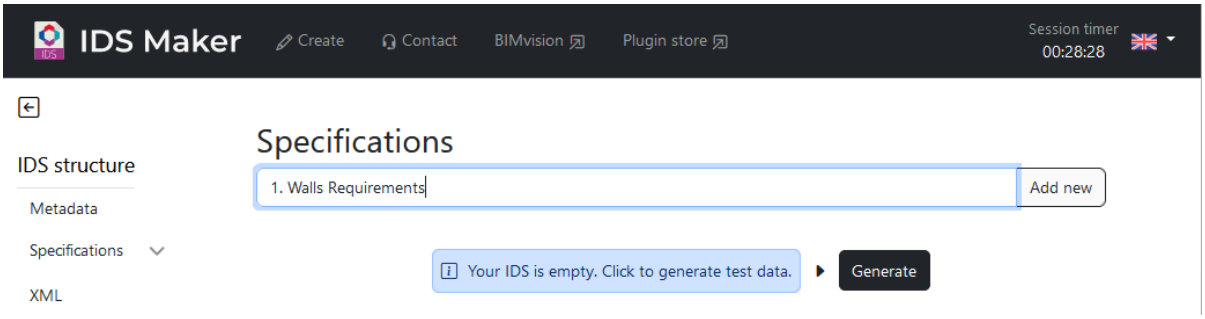


If you generate specifications (or create several sample ones) and want to delete them all to start over, you can click "Delete all".

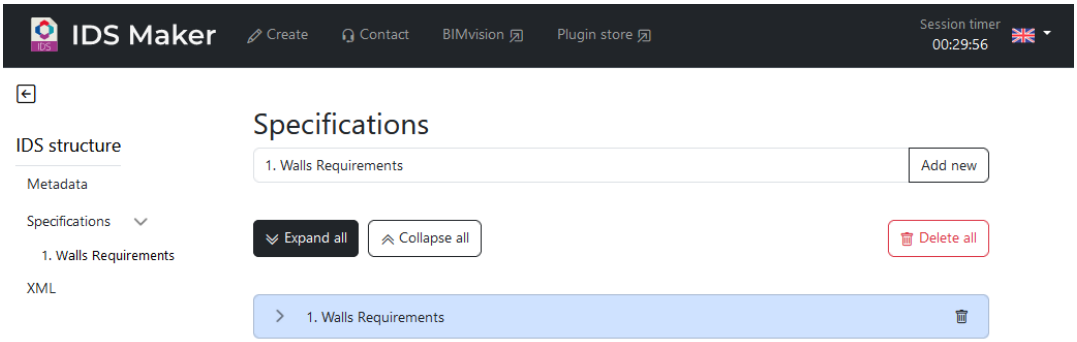


Creating a single specification

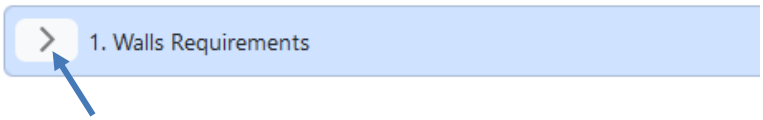
To create a specification, first enter its name and then click **Add new**.



After entering the specification, new buttons will appear: **Expand all**, **Collapse all**, **Delete all** and a bar with the name and access to the specification entered by the user.



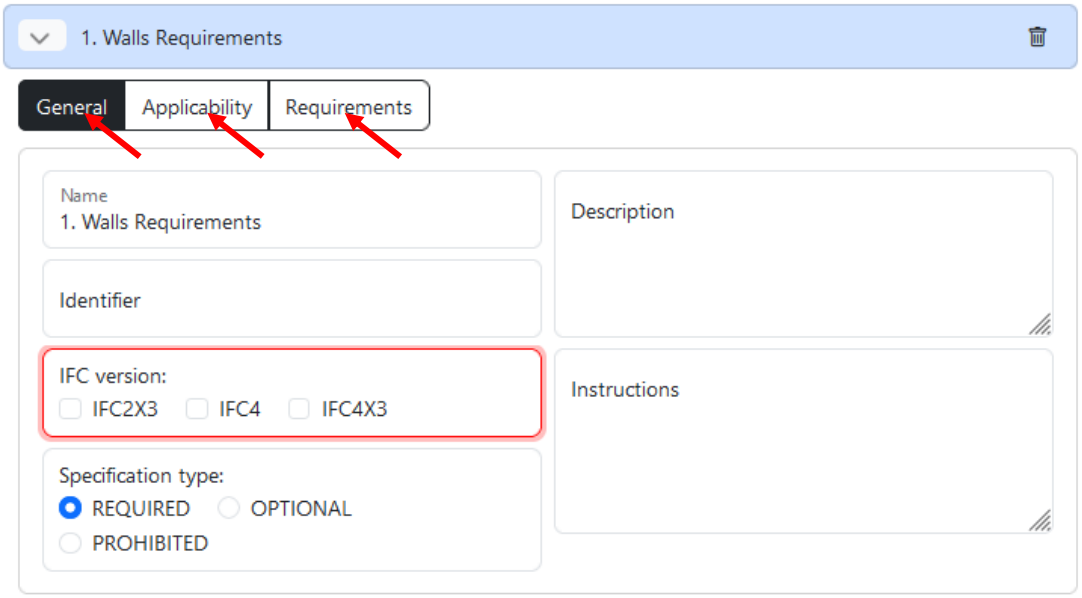
Click on the arrow to the left of the bar to expand the specification content so that you can complete the data and define it.



Each specification has specific parts that need to be defined. These are:

- **Description** (Description data) – in the **General** tab
- **Applicability**
- **Requirements**

By clicking on the individual tabs, we move to the next three sections of the specification.



The "General" tab contains information such as:

- **Name** – the field will display the name entered when adding the specification. You can modify it here at any time
- **Identifier**
- **IFC Version** – required field; specification may apply to one or more versions of the IFC file schema
- **Specification Type** - specification type determines what type of specification it is and directly applies to elements specified in the "Applicability" tab:
    - Required – elements specified in "Applicability" must appear in the model and they must meet the conditions defined in "Requirements"
    - Optional – elements specified in "Applicability" may appear in the model. If they do, then they must comply with the conditions in "Requirements". If the elements specified in "Applicability" do not appear at all, this will also be correct.
    - Prohibited – elements specified in "Applicability" cannot appear in the model
- **Description** – a description of the specification, why it is important, what it is for, etc.
- **Instructions** – instructions for entering information related to this specification (who, when and how should enter the required information in the IFC model)

Applicability tab

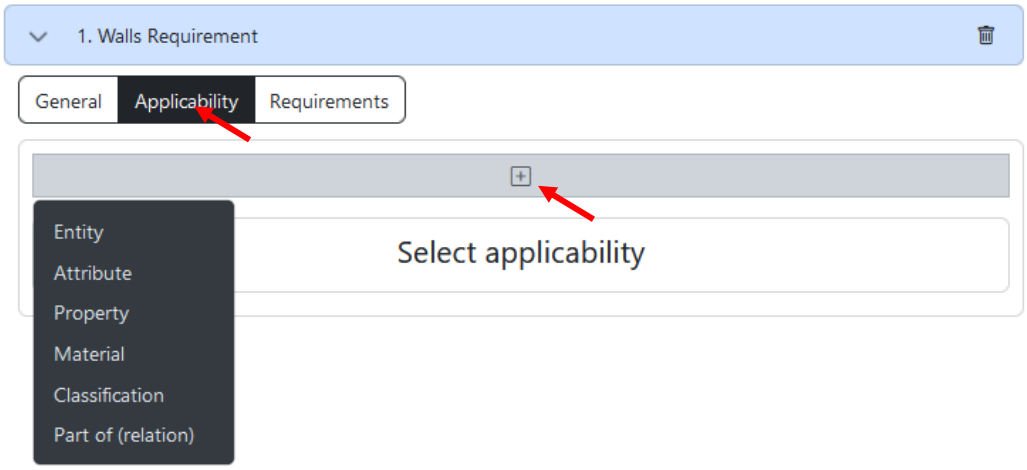
In "Applicability" the elements to which the specification applies are defined. They can be indicated by 6 different categories of information resources - "Facets":

- Entity – defined by IFC class/product type and predefined type,
- Attribute – defined by its name and its value,
- Classification – defined by the system and its value
- Property – defined by the name of the property set it belongs to, the property name and its value
- Material – defined by value

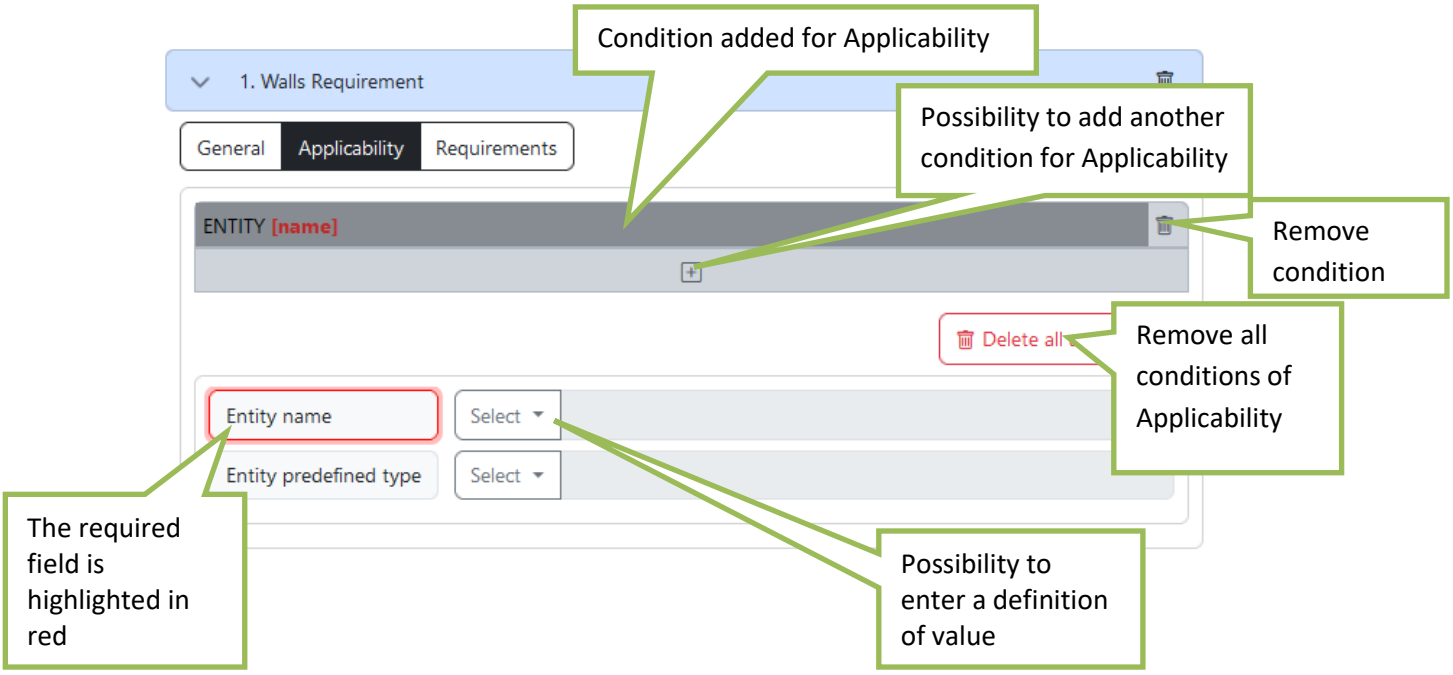
- Parts – defined by the entity and the relationship

Applicability has one overriding condition: MUST (**Required** Specification), MAY (**Optional** Specification), or MUST NOT (**Prohibited** Specification). The specification type, that one overriding condition, is defined in the "General" tab and affects the scope of applicability of the specification.

If the Applicability has more than one condition, the logical condition between them is "and".



After clicking the “+” sign, you must select one of the six possible "facets". Depending on the selected category of information resource, slightly different fields will appear to fill in. Below you can see what it looks like for the "Entity" facet.

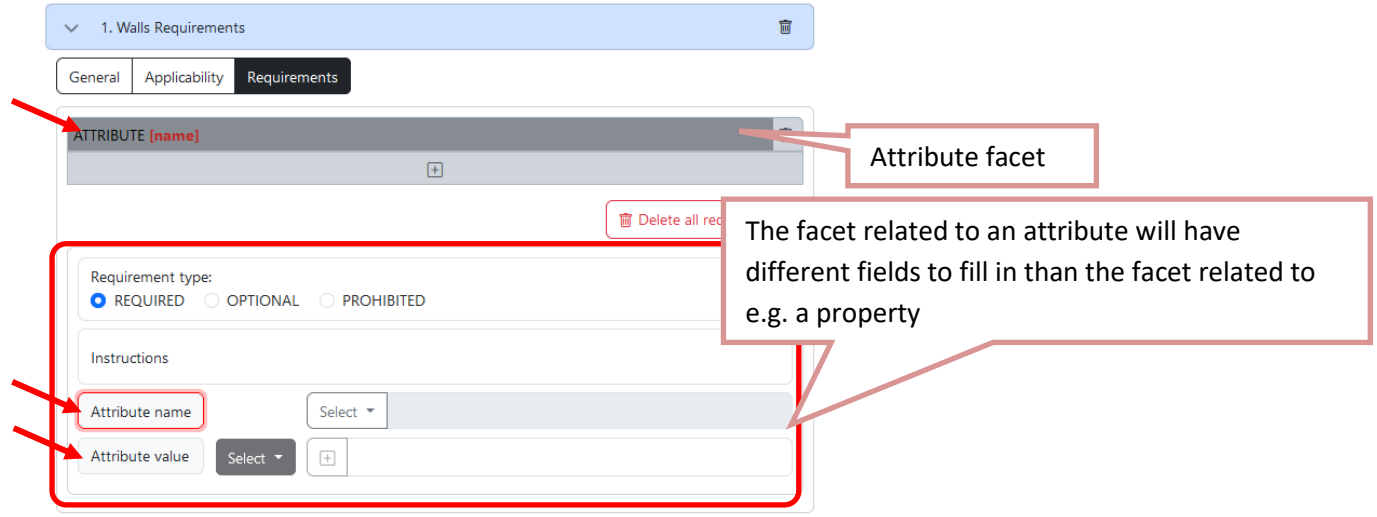


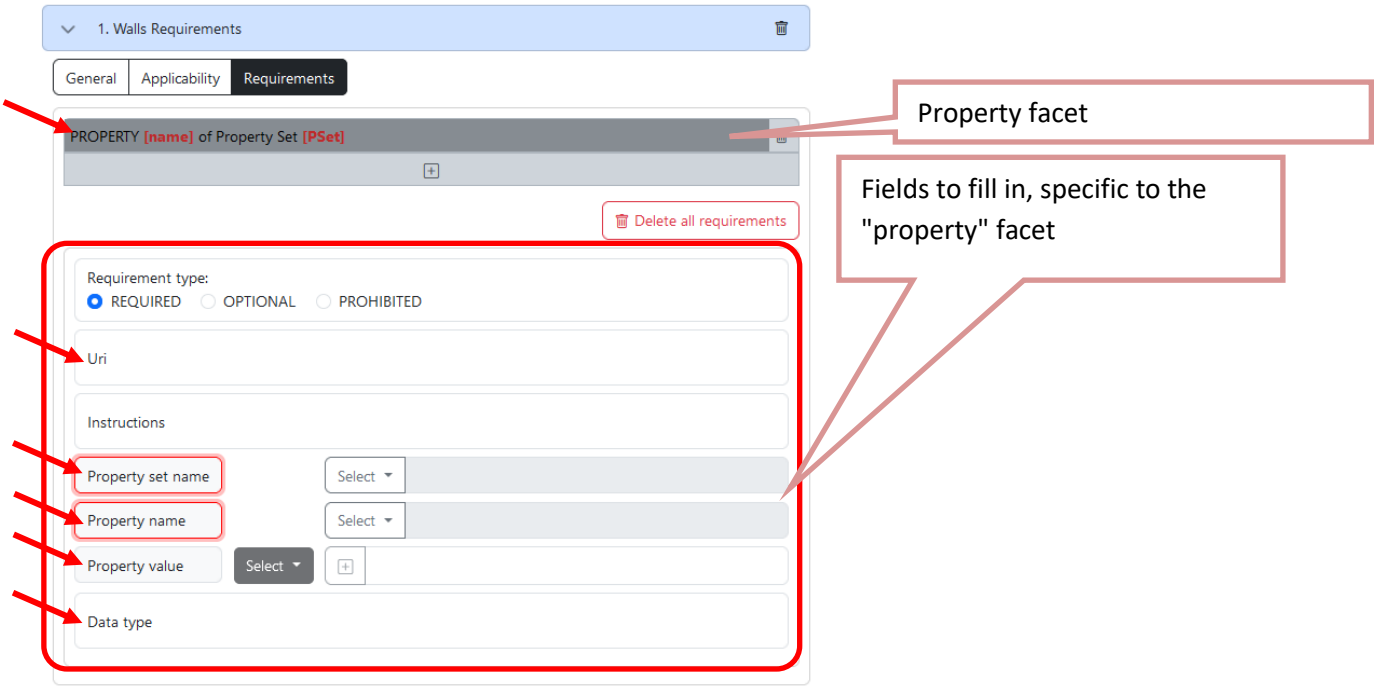
Requirements tab

In "Requirements" individual conditions are defined, which elements defined by "Applicability" must meet. “Requirements” are also (similarly to "Applicability") defined based on "facets". However, each requirement is individual and has its own type. When adding a requirement, we indicate in relation to which information area (facet) we will define it. After this selection, fields to fill in appear.

Each condition must have a specific type (required, optional, prohibited). Property conditions also have a field where you can fill in a URI - that is, a reference to bsDD. Conditions for different facets may have slightly different fields to fill in.

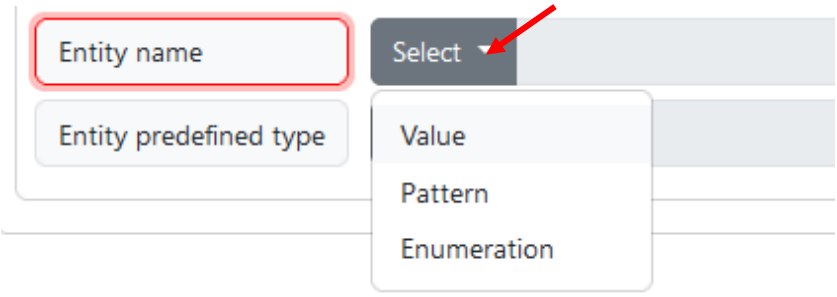
Conditions and their requirements can be defined in different ways. A condition can concern, for example, only having a certain attribute, but it can also concern the value that the attribute should contain.





Defining conditions in specifications

Both when defining the area of Applicability and when creating individual conditions, we may have to deal with various possibilities when entering value definitions.



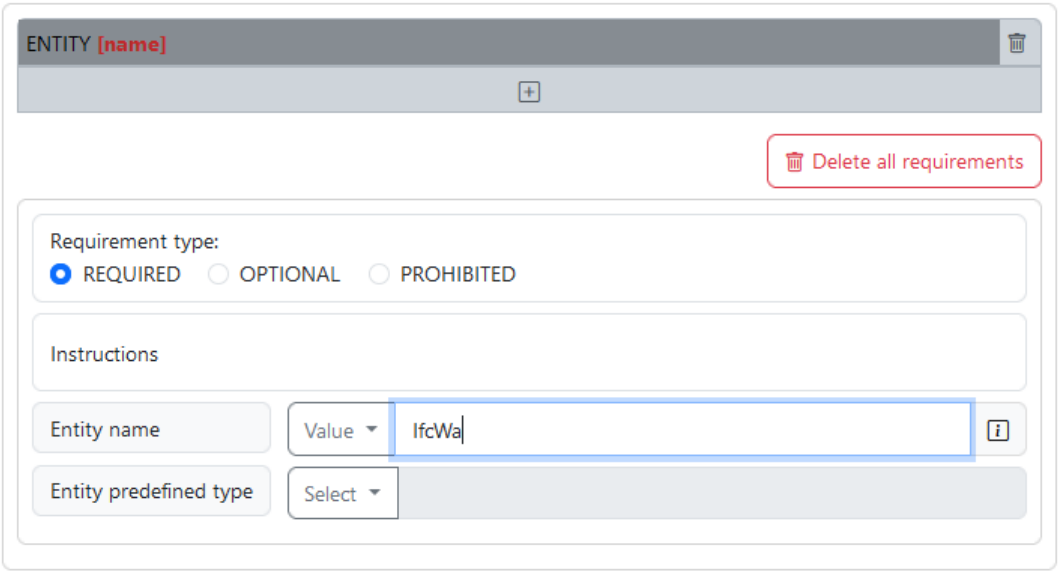
Depending on the facet, the fields to be completed will be different.

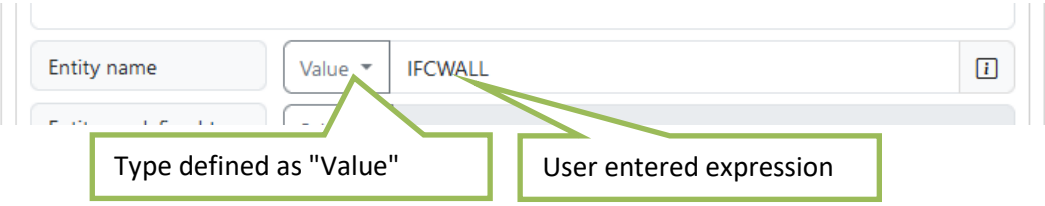
When defining names of properties, entities, attributes, etc., we choose one of three options:

- **Value** – the exact value of the expression (e.g. Entity name defined as equal to the value of "IfcWall")

- **Pattern** – a value defined by a regular expression (a regular expression, also known as Regex, allows you to write a pattern according to which strings of characters are compared) (e.g. if someone wanted the project name to always start with a three-digit number, then contain an underscore, then 3 capital letters indicating the industry, an underscore and any string of characters – the pattern expressing such a value would look like this: ^\d{3}_[A-Z]{3}_.*)

- **Enumeration** – allows you to save a set of possible values (e.g. if you want to define that the value of the property defining the concrete class can only take 3 specific values, you can select this option and then write them out separated by semicolons).

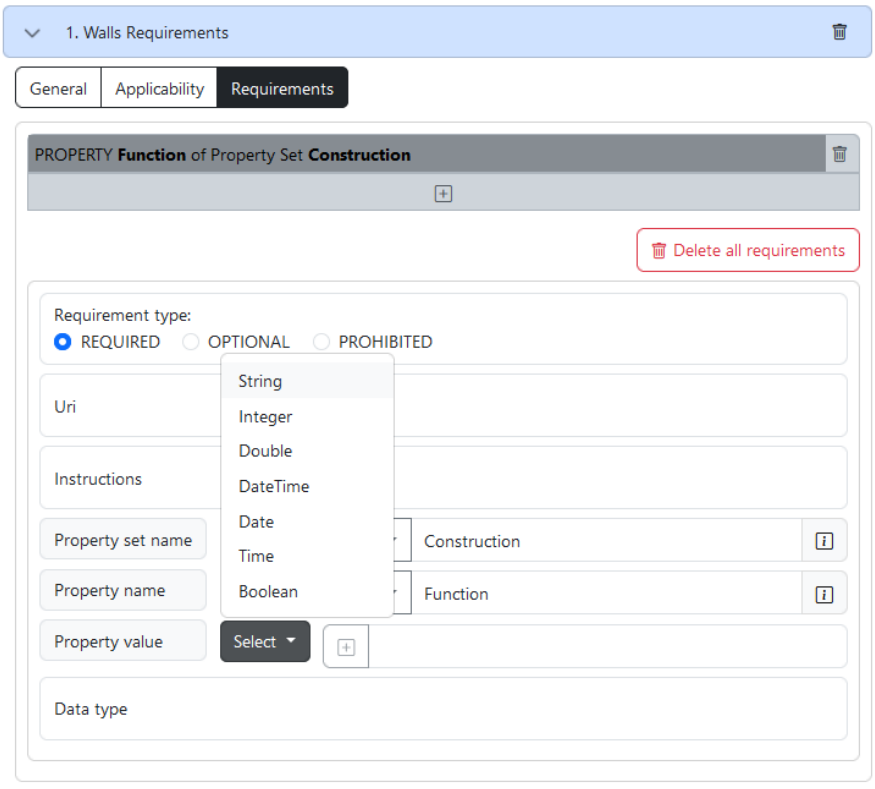


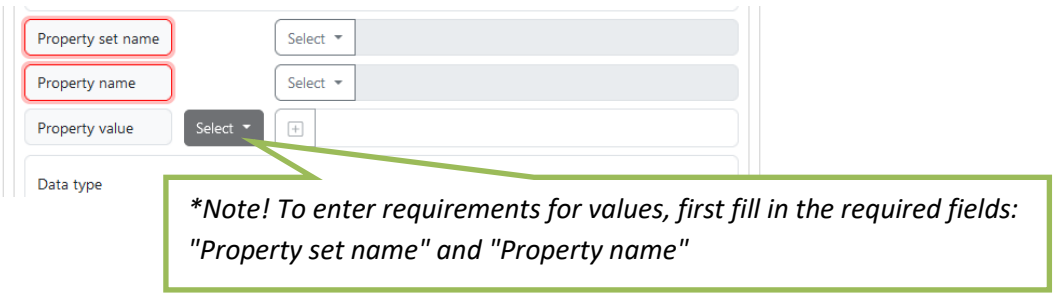


In the case of properties and attributes, the user can also create a requirement regarding requirements related to the value of the specified property or attribute.

For these types of fields, the user must first define what type of values are expected. They can be defined as:

- String – values that are defined as alphanumeric strings
- Integer – whole numbers
- Double – floating-point numbers
- DateTime – format defining a date and time value
- Date – format of a value defining a date
- Time – format defining a time
- Boolean – values defined as “yes” or “no” (true/false)

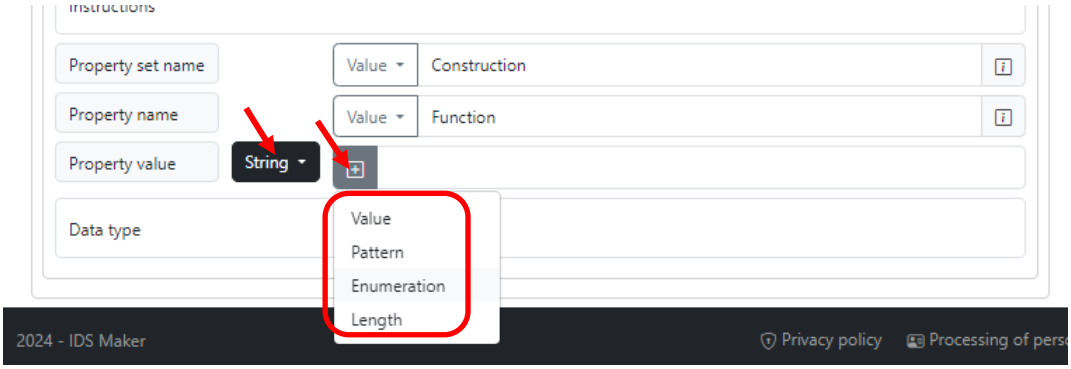




Once we define the type of the value, we can enter a specific definition for the value.

These definitions will be different for different types of these values (string/integer/double…). In the case of values expressed e.g. by string, there are 4 ways to do it:

- Value – by entering the exact expected value
- Pattern – by using a regular expression (Regex)
- List – by entering the possibilities that may appear in this value
- Length – by specifying the number of characters



For example, in the case of numeric values such as integer or double, these will be the options:

- Value – the expected value in the form of the exact expected value

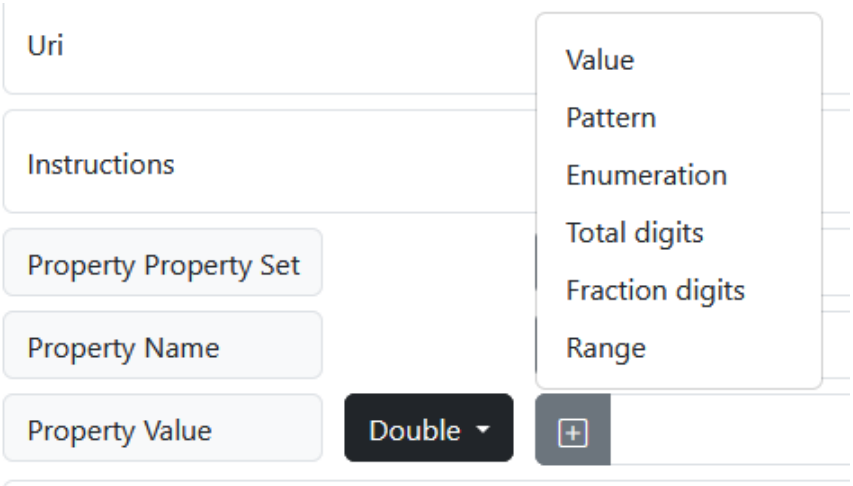
- Pattern – the value specified by a regular expression (regex)

- Enumeration – possible options

- Total digits – specified expression length

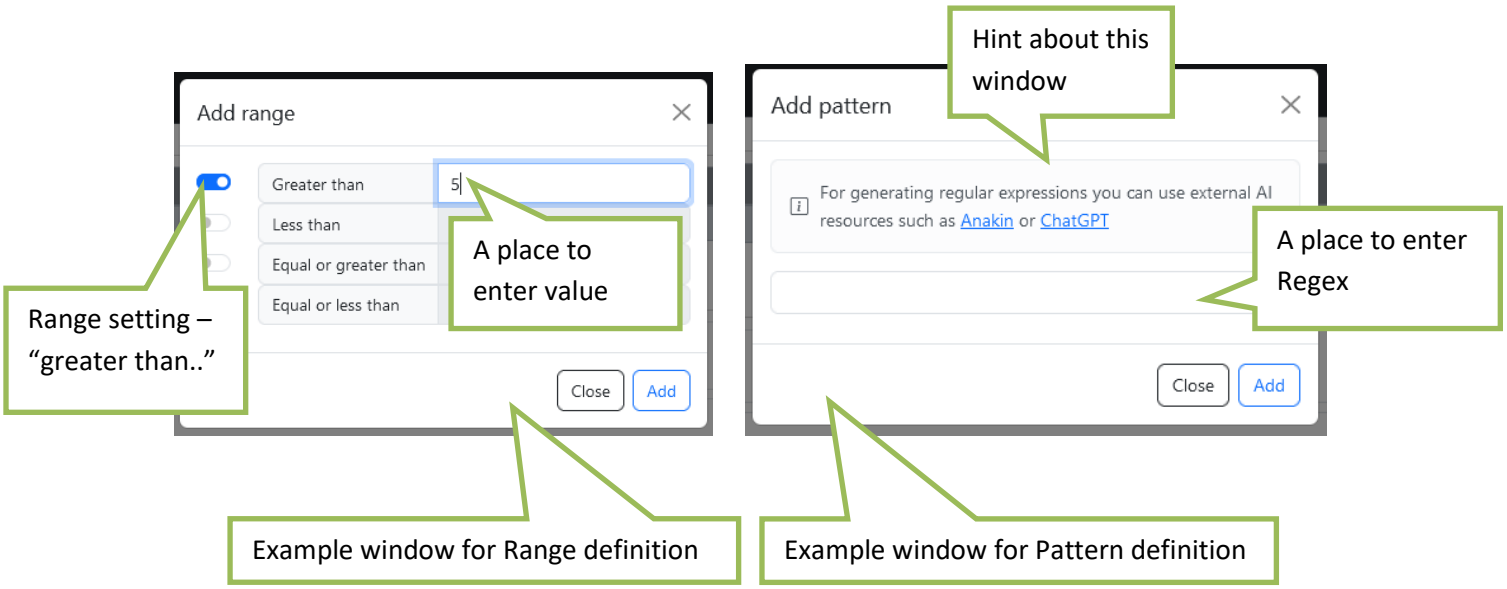
- Fraction digits – specified number of digits after the decimal point

- Range – specified the size of the value by the minimum/maximum range



After selecting one of the available options, an additional window will appear.

In the case of specifying the Range value, you can enable some of the sliders in the window and fill in the corresponding fields, e.g. in the case of a Pattern, you can enter Regex. The fields can also contain [i] additional hints (as in the case of a Pattern, where the hint concerns the possibility of using external AI resources to generate the expected Regex).



Specification examples

** The example focuses on definitions related to specification type, application and conditions. Descriptive parts have been omitted to focus on the most important things in terms of understanding the creation of conditions in IDS files and the subsequent automation of IFC model verification using them with dedicated tools (such as IDS Checker - a solution for BIMvision).*

The example specification would require that the model have elements defined as IFCWALLSTANDARDCASE, and that they have a NetVolume property in the BaseQuantities property set, a Width property in the BaseQuantities property set, a Height property value from the BaseQuantities property set with a value greater than 2.5, a FireRating property from the Pset_WallCommon property set with a value of REI 50 or REI 120, and a Name attribute value defined as "Wall -" followed by either "EXT" or "INT", and finally the " - " may be followed by "L" or "T".

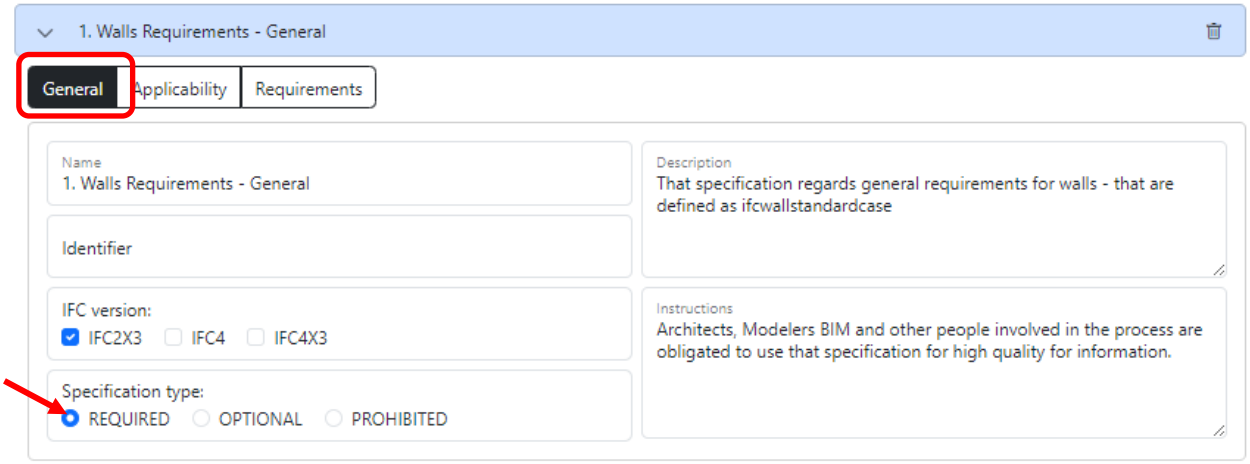
➔ We can interpret this as follows: the model must contain elements defined as IFCWALLSTANDARDCASE that have to meet several different conditions relating to their selected attributes and properties.

Let's break down the definition of such an IDS file into individual steps.

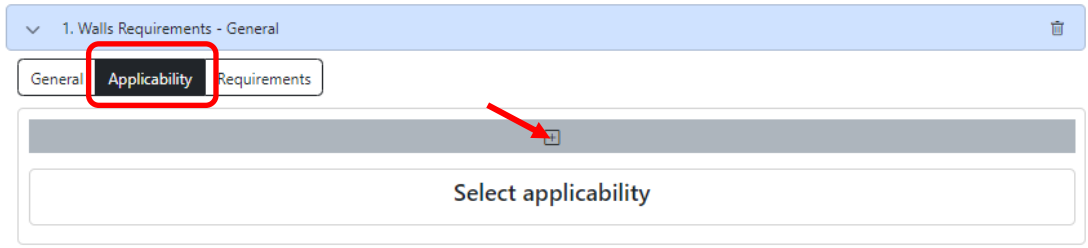
Specification type and Applicability determination

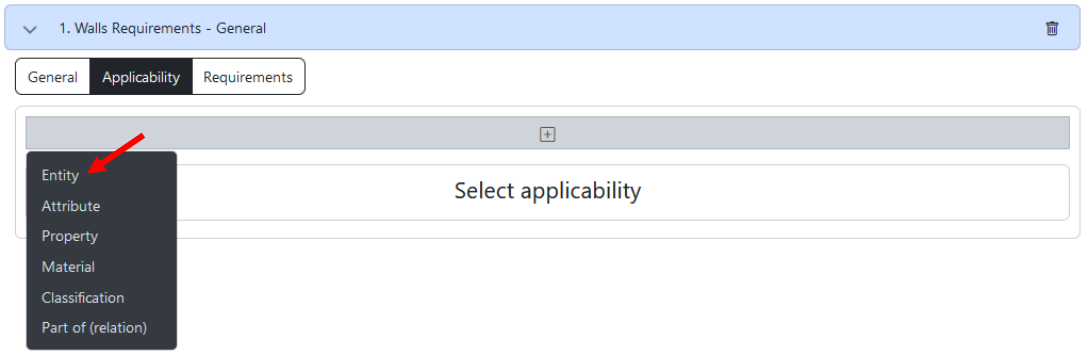
The specification requires that "the model must have elements defined as IFCWALLSTANDARDCASE" -> the model must contain (REQUIRED) elements that have an IFC type equal to IFCWALLSTANDARDCASE (Applicability).

General tab - it is necessary to specify the IFC version (e.g. IFC 2x3), and then, after filling in fields such as: name, description, etc., define the Specification type as REQUIRED

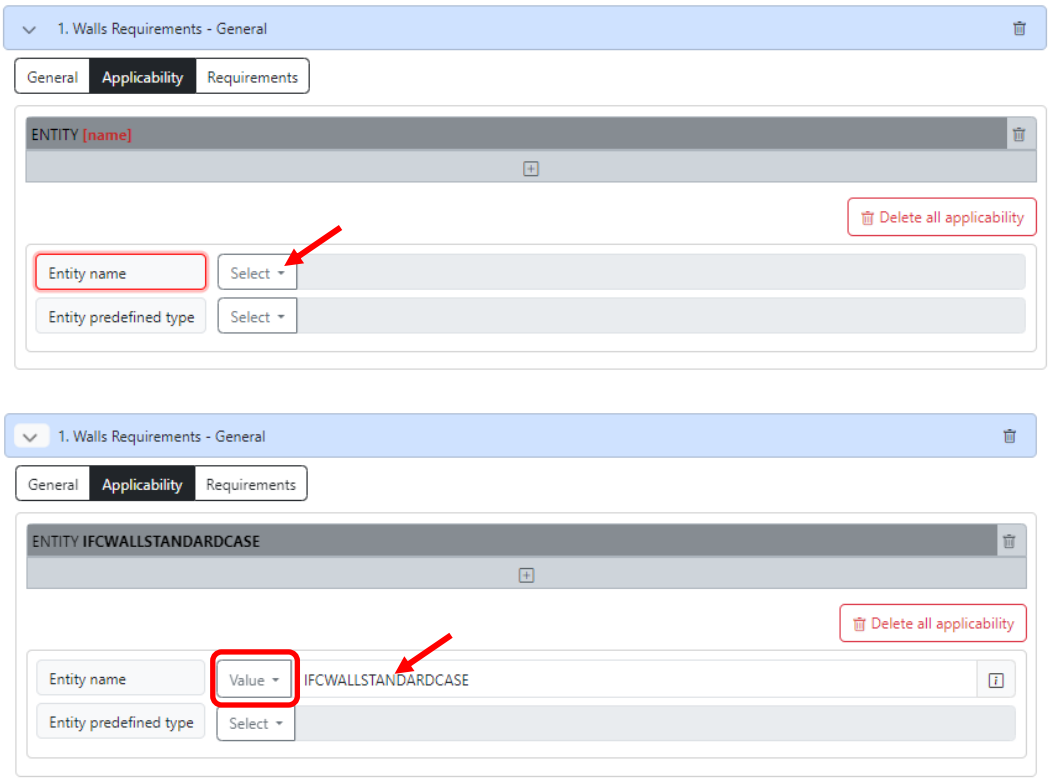


Go to the second Applicability tab and specify that elements with Entity equal to IFCWALLSTANDARDCASE should appear in the model.

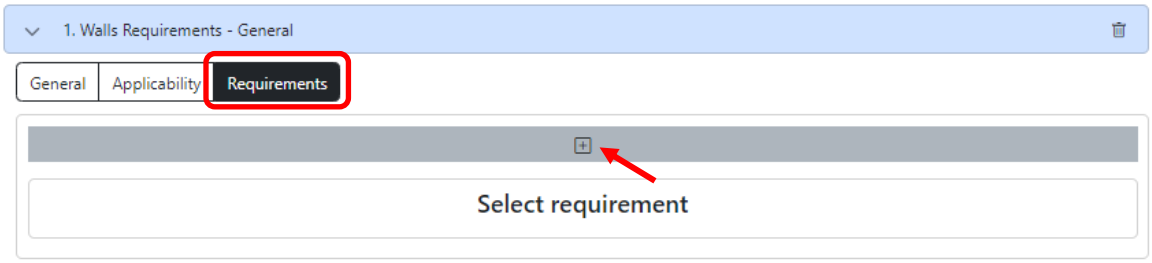




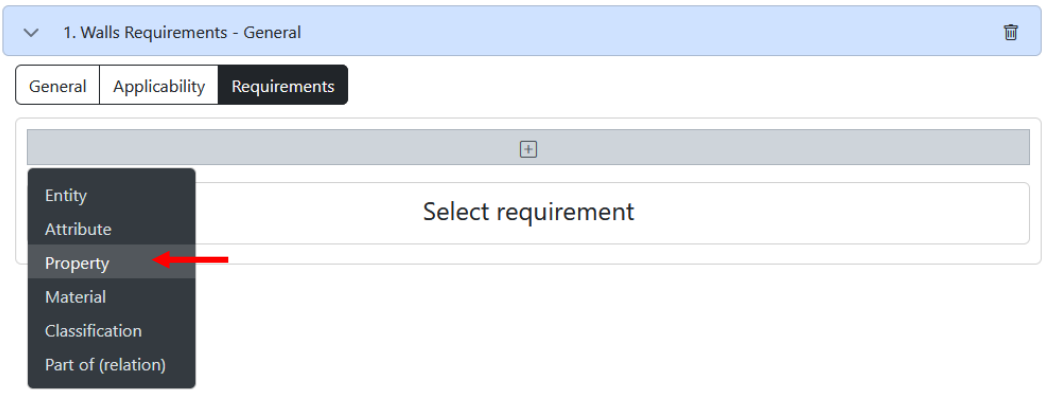
After specifying that we will define the Applicability by "Entity", dedicated fields will appear. It is not necessary to fill all of them. In this example, we will specify that the specification applies to elements with an entity name equal to the value (Value) IFCWALLSTANDARDCASE.

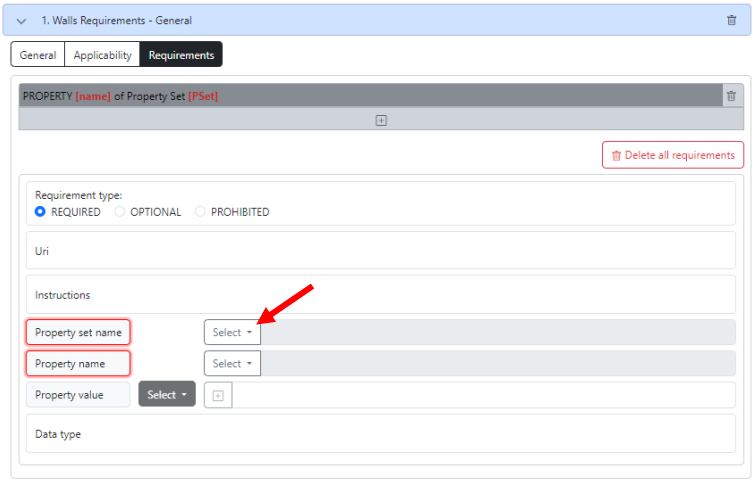


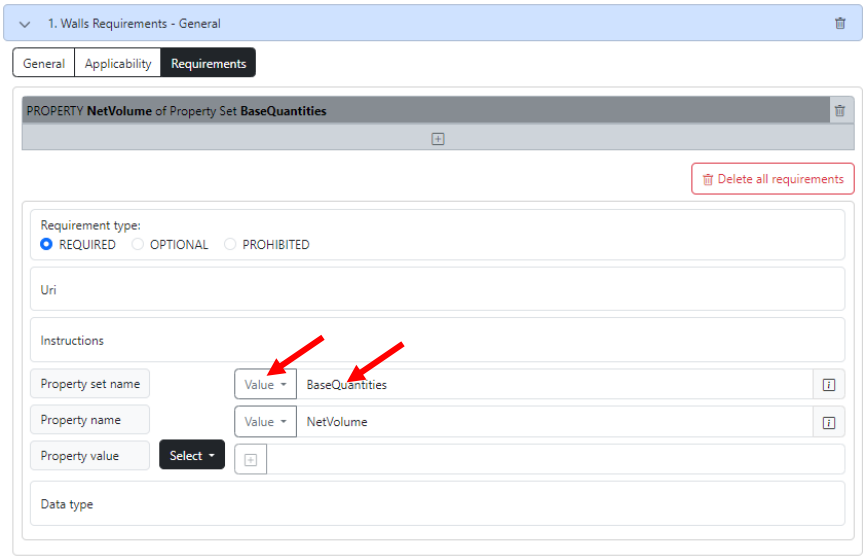
Once the "Applicability" is defined (via the specification type on the "General" tab and the conditions on the "Applicability" tab), you can move on to formulating the conditions that these elements must meet. The conditions for these walls are defined on the “Requirements” tab.



The first requirement concerns properties. It states that items must have a NetVolume property in the BaseQuantities property set.

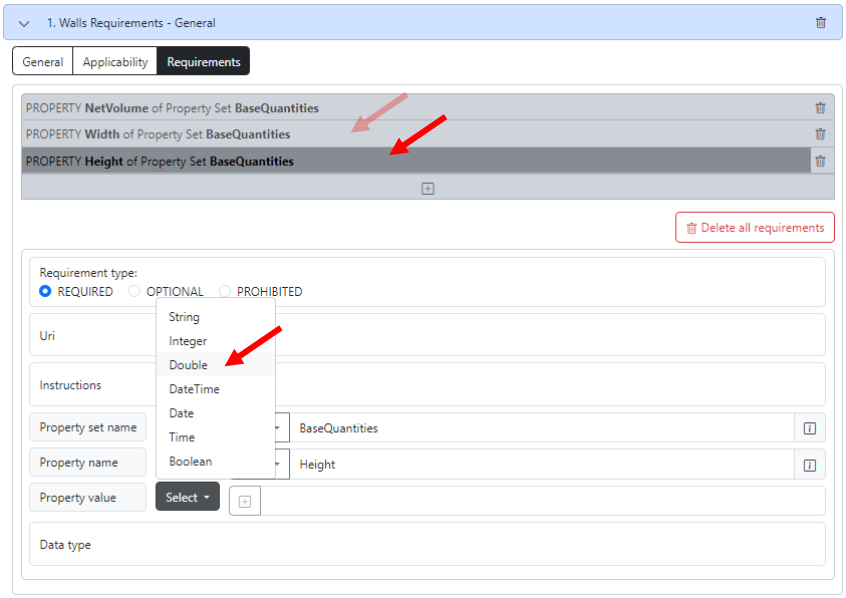




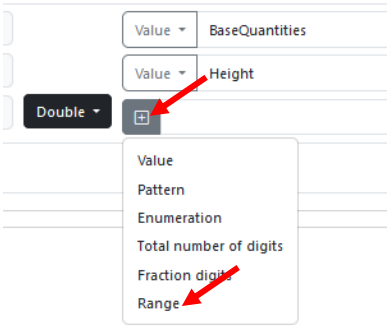


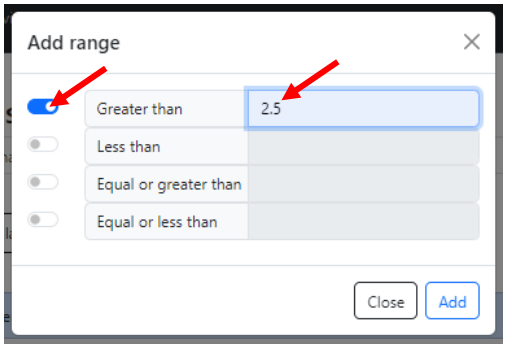
The next requirement is also that the elements must have a specific property (the Width property in the BaseQuantities Property Set). As with the first requirement, a property condition has been added in the same Requirements tab, the "Required" type has been defined, and the property name and property set have been entered in the required fields.

The next requirement is that the specified property has an appropriate value. After adding a condition by property, specifying the type and entering the name and set of properties and properties, you must start specifying the value. To do this, you must select the appropriate format/data type from the drop-down list. The height will be a real number, so you should select "Double".



Next, the user can specify how the value will be defined. In this example, since the requirement is that the height is greater than 2.5, the "+" button should be selected, and then the "Range" option should be selected. In the Range window, "Greater than" should be activated and the value entered should be 2.5.

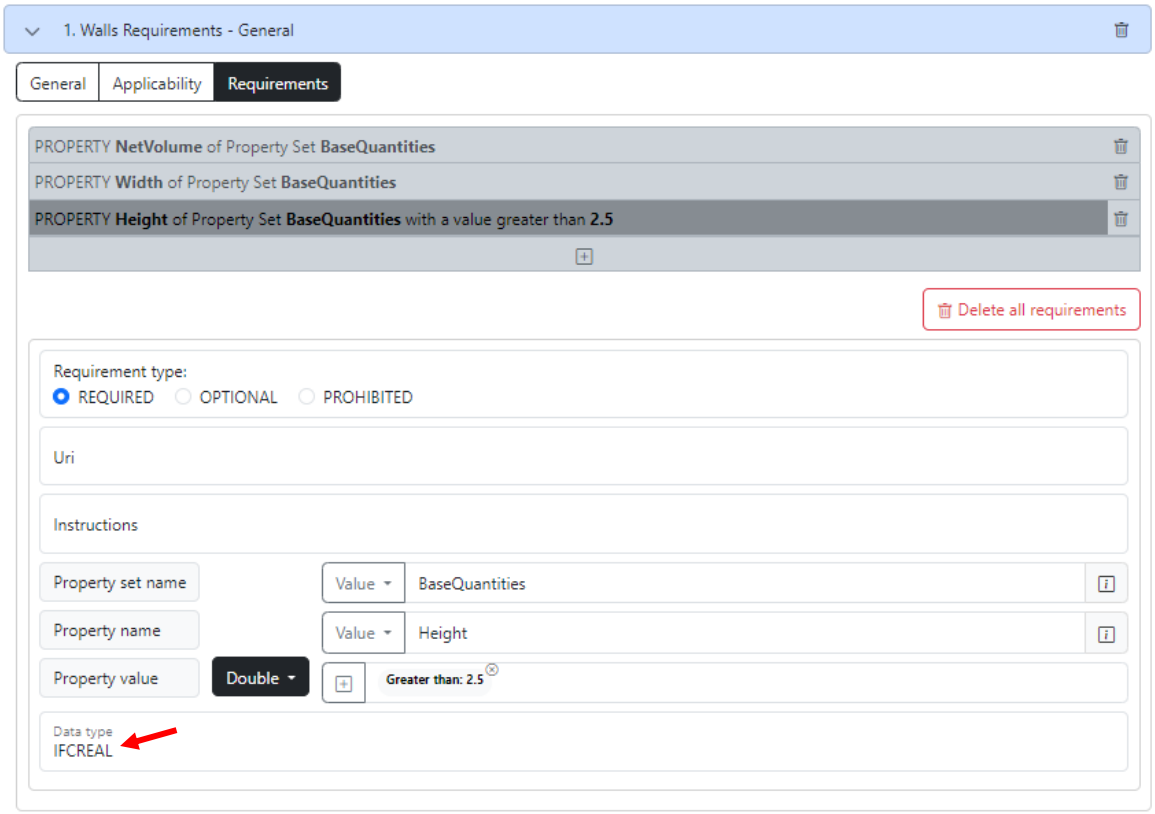




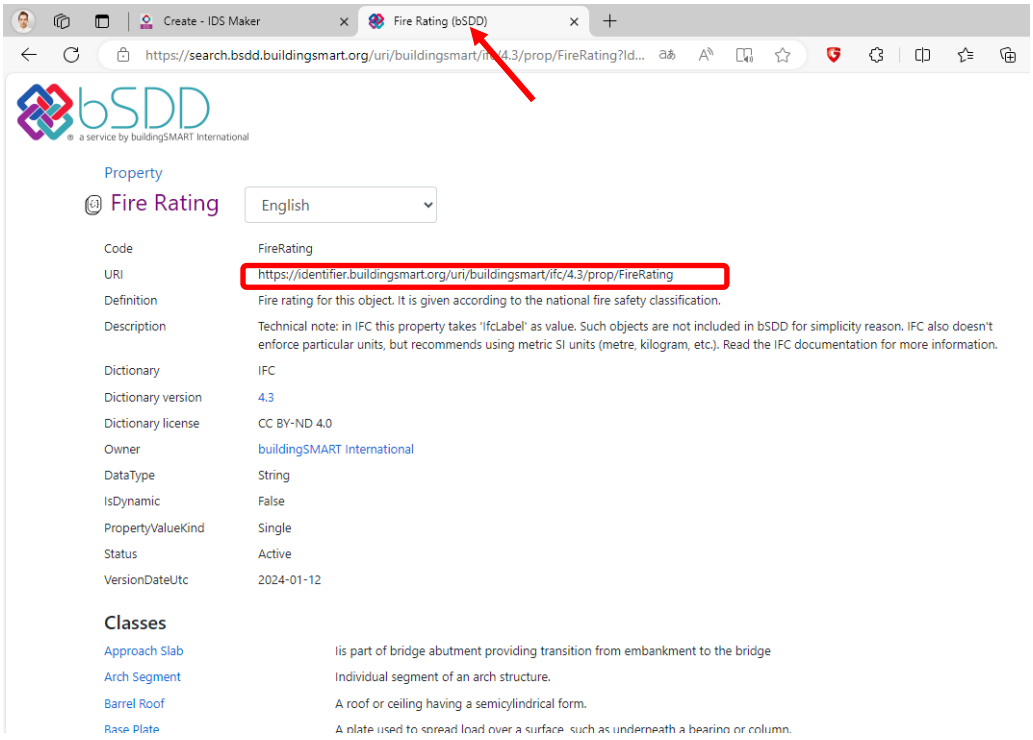
If you are defining a requirement for a property value (not just that a property must be present), you must also enter a "Data type" for the property. The data type is constant for each property. In many cases, the data type for alphanumeric values will be IFCLABEL, and for numeric values IFCREAL or IFCINTEGER for integers, but this is not always the case, as specific properties have their own dedicated data types.

For example, it is worth seeing the table under the indicated link and paying attention to e.g. the ThermalTransmittance property, whose Data type for a specific IFC 2x3 scheme may be equal to IfcThermalTransmittanceMeasure or THERMALTRANSMITTANCEUNIT: https://standards.buildingsmart.org/IFC/RELEASE/IFC2x3/TC1/HTML/psd/IfcSharedBldgElements/Pset_WallCommon.xml

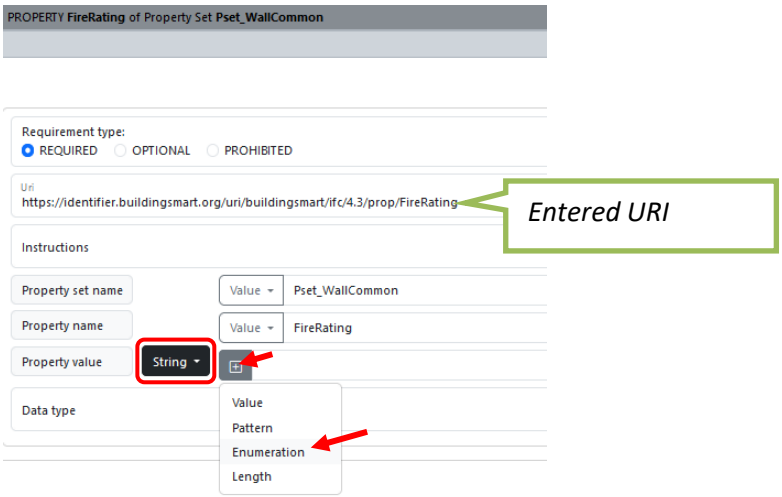
When the user starts entering the Data type, a drop-down list of suggestions will appear.

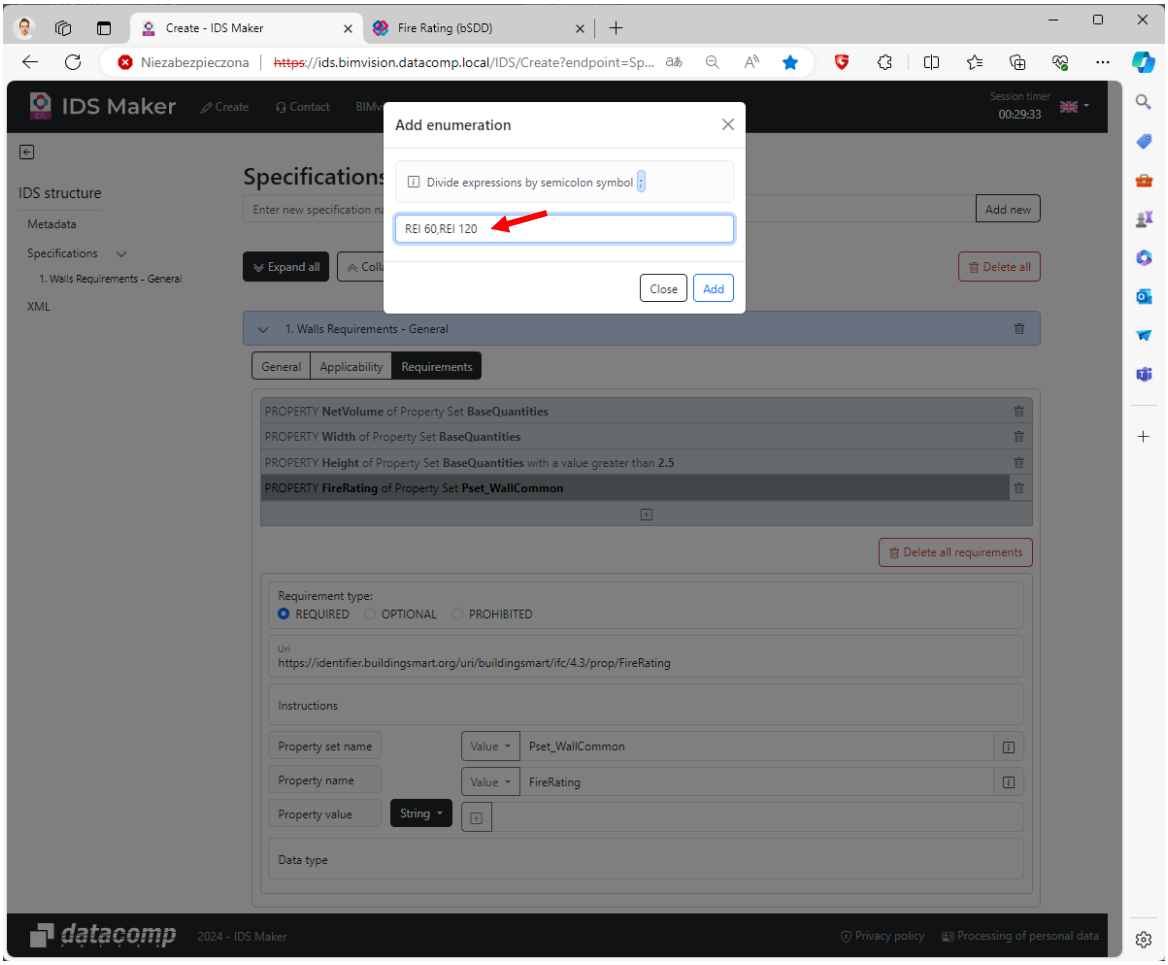


The penultimate condition concerns fire resistance expressed by the requirement for the FireRating property from the Pset_WallCommon property set. For example, for this condition we will also add an URI. And to do this, we need to find the property we are interested in in the external bsDD service. When we find it, we can copy the given URI and paste it into the dedicated field.

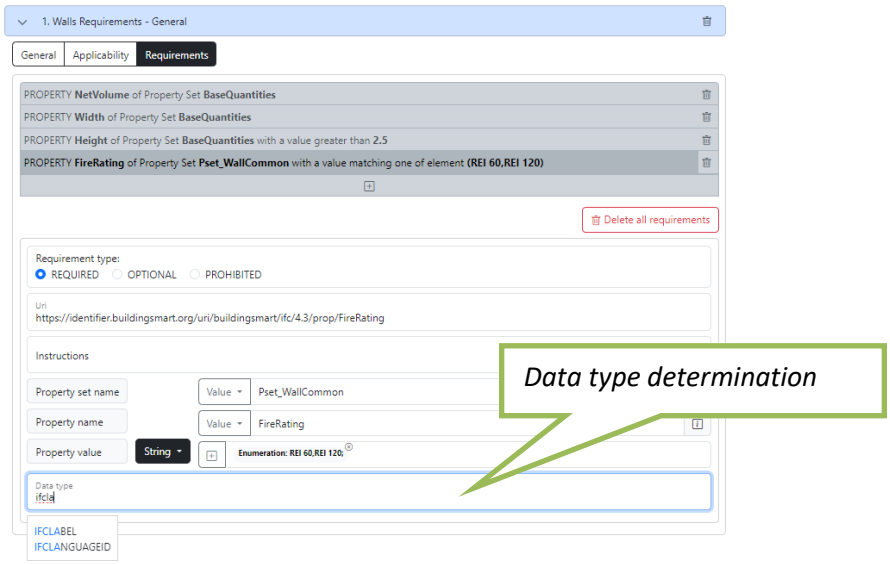


For the FireRating property from the Pset_WallCommon property set, we will specify that the value for the element must be defined as REI 50 or REI 120. To do this, we will select the Enumeration option. We enter the allowable values using a semicolon (without spaces).

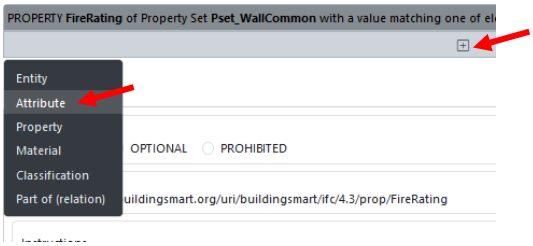




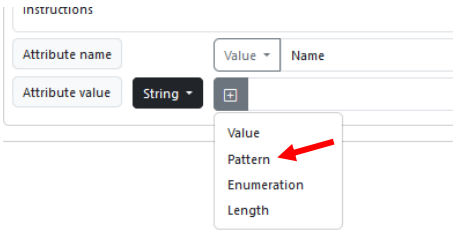
After entering Data type you can proceed to the last condition definition of this sample specification.

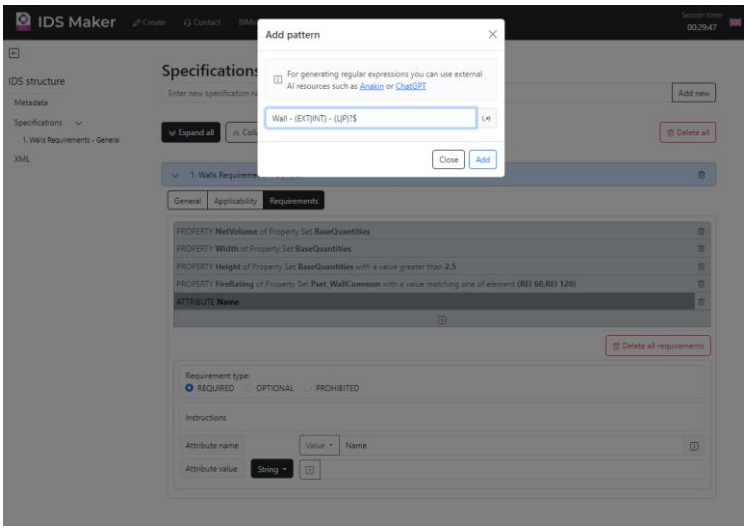


The last requirement is about an attribute. After adding a condition with a reference to an attribute (not a property this time), we can specify that the requirement will be about the "Name" attribute.

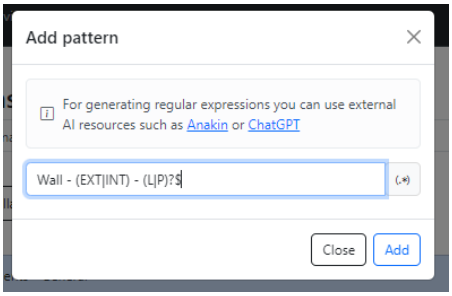


The value will be expressed as a Pattern. After selecting this option, you need to specify it via Regex (regular expression).





The individual walls should have appropriately defined names. The first part should be equal to "Wall -" then there should be a definition of whether it is external or internal, using the abbreviation "EXT" or "INT", and at the end after " - " there may be an additional designation in the form of "L" or "T". The pattern is written using a regular expression (Regex), which is a specific - standardized notation of this type of expression. Regex can define characters in a string, e.g. by specifying the number of consecutive digits, letters or indicating what specific words or characters are or may be included in the expression.



After adding this definition, an example specification was created where the "IFCWALLSTANDARDCASE" walls were required to meet 5 different requirements.

Once you have filled in the rest of the metadata information and any additional information you would like to include, you can go to “XML” on the left side of the site and download your IDS file with the specification you have created. You can also create more specifications so that your IDS also addresses other elements and aspects.

The above specification is an example. You can formulate the specifications in a different way, e.g. by dividing them into narrower topics (e.g. fire resistance range, material data for walls, information for walls - stage 1 or something else).

You can also create your IDS file from simpler specifications with fewer conditions (so that the conditions are in individual specifications).

Release notes

1.0

Contact and Copyright

For questions or suggestions, please contact us at: contact@bimvision.eu; please include the following text in the title: **IDS Maker**.

BIMvision plugins: store.bimvision.eu