

Introduction to Matrix

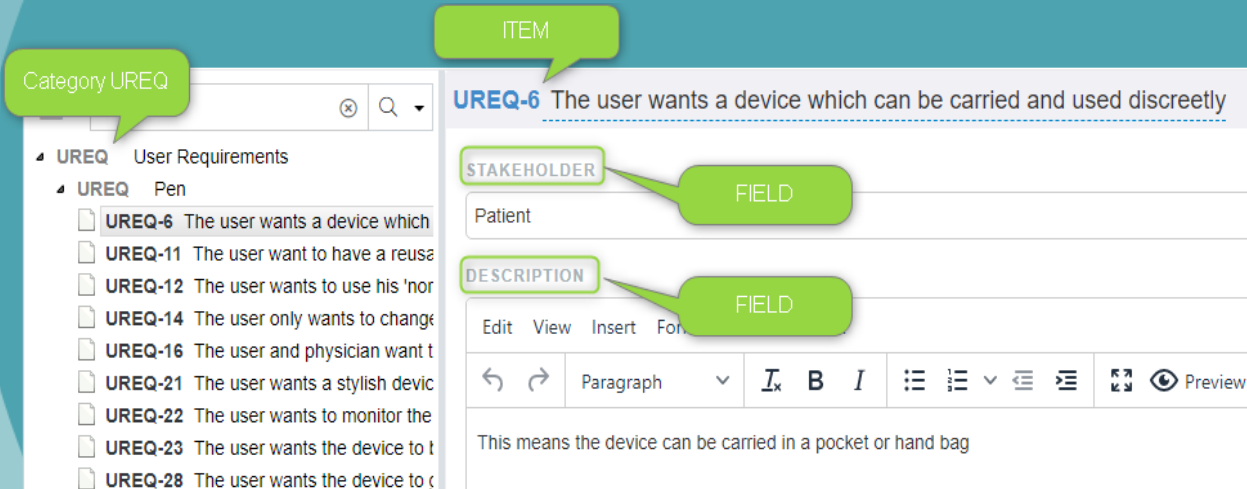
What do we call things?



ITEMS, FIELDS & CATEGORIES

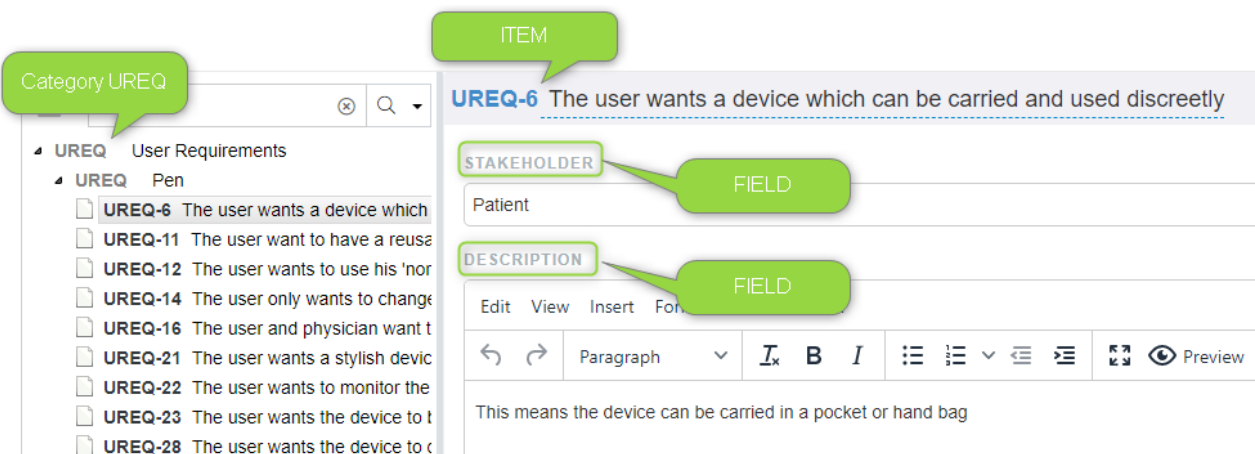
Matrix is designed to store pieces of structured information in a database.

- Each piece of information we call an ITEM
- Each item consists of several FIELDS
- A CATEGORY defines which fields exist in an item and groups them together



The screenshot illustrates the Matrix Requirements software interface. On the left, a tree view shows a category named 'UREQ' (User Requirements) containing several items, including 'UREQ-6 The user wants a device which can be carried and used discreetly'. A green callout bubble labeled 'Category UREQ' points to the category name. On the right, the detailed view of 'UREQ-6' is shown. It features a title bar with the item name and description. Below the title, there are two fields: 'STAKEHOLDER' with the value 'Patient' and 'DESCRIPTION' with the value 'This means the device can be carried in a pocket or hand bag'. Green callout bubbles labeled 'ITEM' and 'FIELD' point to the title bar and the fields respectively. The interface also includes a search bar, a list of items, and a rich text editor with various formatting options.

ITEMS, FIELDS & CATEGORIES



The screenshot shows the Matrix Requirements software interface. On the left, a tree view displays a hierarchy of user requirements under the category 'UREQ User Requirements'. The selected item is 'UREQ-6 The user wants a device which can be carried and used discreetly'. The main view shows the details for this item, including a 'STAKEHOLDER' field with the value 'Patient' and a 'DESCRIPTION' field with the text 'This means the device can be carried in a pocket or hand bag'. The interface includes a search bar, a list of requirements, and a rich text editor for the description.

🕒 You might have a category, for **User Requirements**, which

- Is defined using a **unique prefix** for all user requirements, say UREQ
- Is structured using different **fields** to **detail** each user requirement, for example, with a drop-down menu for the stakeholder and a description

🕒 The database will have several user requirements for your product

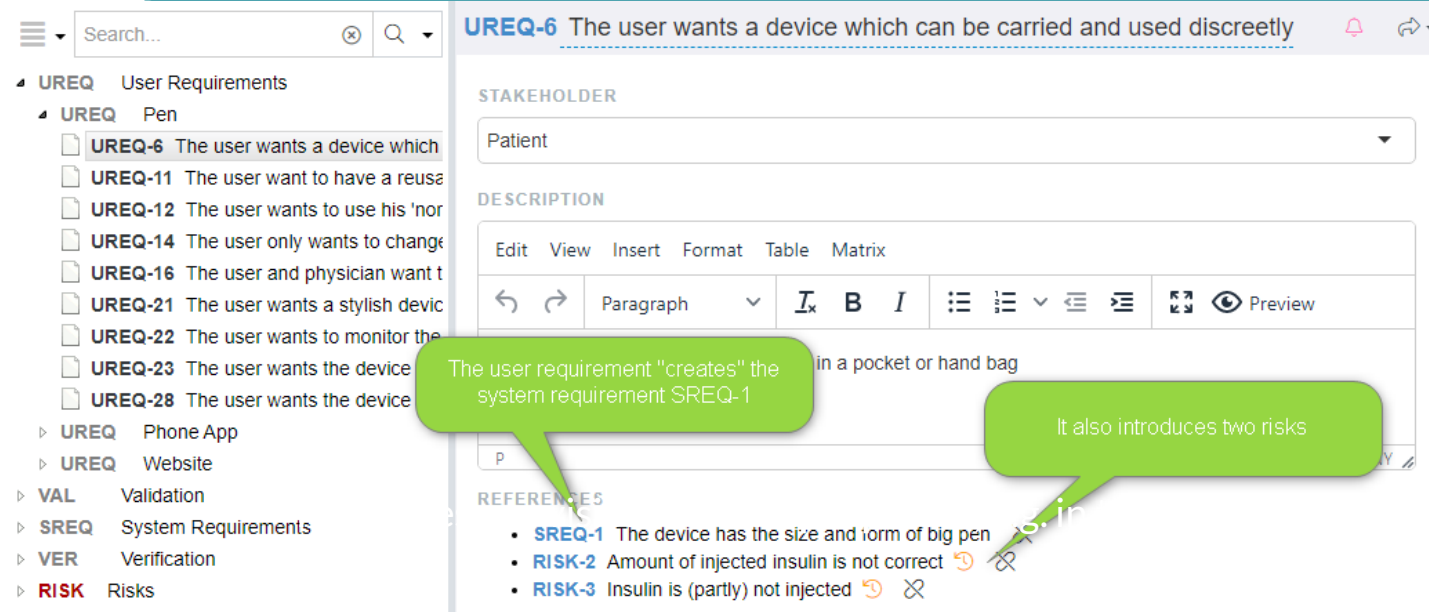
UREQ-1, UREQ-2, UREQ-3, ...

🕒 These are stored in a tree structure to allow you to easily group and find them

TRACEABILITY

- In a typical database you have many different categories with relationships between them
- These are expressed as TRACEABILITY rules
- There are UP TRACES and DOWN TRACES

- The up or down designates the ‘direction’ of a trace, e.g., if you have a user requirement you will need to break it down into system requirements.
- In Matrix, you would for example define a trace rule that “a UREQ must have a trace to SREQ”. UREQ → SREQ is therefore a down trace.
- Usually if there is a **down trace** there is one in other direction which is called an up trace, e.g., “a system requirement is derived from a user requirement” UREQ ← SREQ



The screenshot shows the Matrix Requirements tool interface. On the left, a tree view displays a hierarchy of requirements: UREQ (User Requirements) - Pen, UREQ-6 (The user wants a device which...), UREQ-11 (The user want to have a reuse...), UREQ-12 (The user wants to use his 'nor...), UREQ-14 (The user only wants to change...), UREQ-16 (The user and physician want t...), UREQ-21 (The user wants a stylish devic...), UREQ-22 (The user wants to monitor the...), UREQ-23 (The user wants the device...), UREQ-28 (The user wants the device...), UREQ (Phone App), UREQ (Website), VAL (Validation), SREQ (System Requirements), VER (Verification), and RISK (Risks).

The main panel shows the details for UREQ-6: "The user wants a device which can be carried and used discreetly". The stakeholder is "Patient". The description field contains the text "in a pocket or hand bag". Below the description, the REFERENCES section lists three items: SREQ-1 (The device has the size and form of big pen), RISK-2 (Amount of injected insulin is not correct), and RISK-3 (Insulin is (partly) not injected).

Two green callout boxes provide additional context:

- The first callout points to the description field and states: "The user requirement 'creates' the system requirement SREQ-1".
- The second callout points to the RISK-2 and RISK-3 entries and states: "It also introduces two risks".

DOCUMENTS

- 📌 **Documents** help you to get data from the **database** into **PDFs, Word or Excel**
- 📌 They can be records, plans, reports, minutes, ...
- 📌 Documents are stored in the **CATEGORY DOC**
- 📌 Each Document contains several **SECTIONS** that can be
 - Just some text, a checkbox etc.
 - or the content of **ITEMS** in lists or tables
- 📌 To **freeze the content** of a **DOC** it is converted into **SIGN items** which can be **electronically signed** but otherwise won't change if the database items inside are changed

SUMMARY

CATEGORY → *Defines which fields exist for an item*

ITEM → *A database entry with several fields*

FIELD → *Holds some information as part of an item*

TRACEABILITY → *Links between items*

DOC → *A document to print the content of items*

SIGN → *An electronical signed document*

SECTION → *A specific type of paragraph composing a document*