ReqView
Custom Requirement Attributes

reqview@eccam.com
Requirement Attributes

- **Basic**
  - Unique ID
  - Description
  - Type

- **Planning**
  - Target Release
  - Risk
  - Effort

- **Collaboration**
  - Discussion
  - Status

- **Priority**
  - Urgency
  - Severity

- **Verification**
  - Acceptance Criteria
  - Verification Method
Elementary Attribute Types

- **int** - integer number
- **bool** - logical value, *true* or *false*
- **string** - string value
- **date** - date stored as [ISO 8601](https://en.wikipedia.org/wiki/ISO_8601) string
- **real** - floating point number
- **enum** - enumeration value
- **xhtml** - [XHTML](https://en.wikipedia.org/wiki/XHTML) string
Predefined Requirement Attributes

- **id**: requirement identifier (*string*)
  - Unique in the document
  - Read only and automatically incremented

- **heading**: section name (*string*)
  - Should not be filled for requirements, only for sections (requirements grouped to features)

- **text**: rich text requirement description (*xhtml*)
Predefined Requirement Attributes

• **deleted** : delete flag (bool)
  – If a requirement is deleted then this flag is set to true. Otherwise the property it is not set.

• **discussion** : list of comments (object)
  – Each comment stores author’s *email, name, and company*, comment *date* and *text*
  – Last user’s comment can be modified if is not followed by a comment made by another user.
Predefined Requirement Attributes

- **history**: list of all requirement modifications *(object)*
  - Each modification stores name of the changed attribute, its old value, modification *date*, author’s *email*, *name*, and *company*.
  - History can be cleared to track changes since some project version.
Definition of Elementary Attributes

- attributeName is a chosen unique identifier of the custom attribute, e.g. "status"
- type describes attribute type, e.g. "int", "date", ...
- readOnly is an optional flag set to true if user should be prevented from changing the attribute value

```
<attributeName>: {
    type: <string>,
    name: <string>, // optional
    readOnly: <bool> // optional
},
```
Example: Effort Estimation

• Define an attribute for effort estimation of each attribute.

• Enables computing the effort estimation per a feature (group of requirements) and total effort estimation of the whole product.

```json
"effort": {
  "name": "Effort [MD]",
  "readOnly": false,
  "type": "int"
},
```
Example: Acceptance Criteria

- Define an attribute for rich text description of the requirement acceptance criteria
- Allows to discuss and approve stakeholder requirements together with acceptance tests.

```
"acceptance": {
    "name": "Acceptance",
    "type": "xhtml"
},
```

This is a free text description of the verification method for the requirement:

1. First do this
2. Then do that
3. What to finally check
Example: Assign Requirements to Milestones

- Define a boolean flag for each milestone in the project.
- Set this flag to true if a requirement or a feature shall be delivered in that milestone.
- Enables filtering requirements assigned to a given project milestone.

```
"Rel-1.0": {
  "name": "Release 1.0",
  "type": "bool"
},
```
Definition of Enumeration Attributes

```json
<attributeName>: {
    type: "enum",
    name: <string>, // optional
    values: [  
        { key: <string>, label: <string>, default: <bool> }, ...
    ],
    readOnly: <bool>, // optional
    multiValued: <bool> // optional
}
```

- **multiValued** is an optional flag set to `true` if the attribute can have multiple enumeration values.
- **values** is an array of enumerated values. Each value shall have a unique string `key` and can have optionally a text `label`, and an optional flag `default` denoting the default value.
Example: Requirement State

- Define set of allowed requirement states to manage requirement lifetime

```
"state": {
   "name": "State",
   "type": "enum",
   "values": [
      { "key": "Draft" },
      { "key": "Accepted" },
      { "key": "Rejected" },
      { "key": "Implemented" },
      { "key": "Verified" }
   ]
}
```
Example: Allocation to Components

- Define set system components and assign requirements to one or more components

```
"components": {
  "name": "Components",
  "type": "enum",
  "values": [
    { "key": "ReqView Desktop" },
    { "key": "ReqView Mobile" },
    { "key": "Synchronizer" },
    { "key": "Server" },
    { "key": "Support" }
  ],
  "multiValued": true,
},
```
Document Attributes

In this dialog you can customize attributes for the current document. Edit attributes definition in JSON text format. Attributes definition is validated before changing.
Custom Attributes column summarizes the values of all defined attributes for each requirement.
Customize Attributes View

Reorder or resize attribute columns by drag and drop.

### Custom Requirements Attributes View

<table>
<thead>
<tr>
<th>ID</th>
<th>Description</th>
<th>Components</th>
<th>Status</th>
<th>Priority</th>
<th>Components</th>
<th>Deadline</th>
<th>Effort [MD]</th>
</tr>
</thead>
<tbody>
<tr>
<td>28</td>
<td>3 Features</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>32</td>
<td>3.1 ReqView Desktop</td>
<td>ReqView Desktop</td>
<td>Implemented</td>
<td></td>
<td>ReqView Desktop</td>
<td></td>
<td></td>
</tr>
<tr>
<td>55</td>
<td>3.1.1 File Operations</td>
<td>ReqView Desktop</td>
<td>Implemented</td>
<td></td>
<td>ReqView Desktop</td>
<td></td>
<td></td>
</tr>
<tr>
<td>253</td>
<td>3.1.1.1 Create New File</td>
<td>ReqView Desktop</td>
<td>Implemented</td>
<td></td>
<td>ReqView Desktop</td>
<td></td>
<td></td>
</tr>
<tr>
<td>254</td>
<td>User shall be able to create a new empty file.</td>
<td>ReqView Desktop</td>
<td>Implemented</td>
<td></td>
<td>ReqView Desktop</td>
<td></td>
<td></td>
</tr>
<tr>
<td>255</td>
<td>User shall be able to save the opened document containing unsaved changes when creating a new file.</td>
<td>ReqView Desktop</td>
<td>Implemented</td>
<td></td>
<td>ReqView Desktop</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>
### Change Values of Custom Attributes

<table>
<thead>
<tr>
<th>Custom Attributes</th>
<th>Status</th>
</tr>
</thead>
<tbody>
<tr>
<td>Components:</td>
<td>ReqView Desktop</td>
</tr>
<tr>
<td>Status:</td>
<td>Implemented</td>
</tr>
<tr>
<td>Is Req.:</td>
<td>Optional</td>
</tr>
<tr>
<td>Status:</td>
<td>Unknown</td>
</tr>
<tr>
<td>Components:</td>
<td>ReqView Desktop</td>
</tr>
<tr>
<td>Is Req.:</td>
<td>Optional</td>
</tr>
<tr>
<td>Status:</td>
<td>Urgent</td>
</tr>
<tr>
<td>Components:</td>
<td>ReqView Desktop</td>
</tr>
<tr>
<td>Is Req.:</td>
<td>Optional</td>
</tr>
<tr>
<td>Status:</td>
<td>Optional</td>
</tr>
<tr>
<td>Components:</td>
<td>ReqView Desktop</td>
</tr>
<tr>
<td>Is Req.:</td>
<td>Optional</td>
</tr>
<tr>
<td>Status:</td>
<td>Optional</td>
</tr>
<tr>
<td>Components:</td>
<td>ReqView Desktop</td>
</tr>
<tr>
<td>Is Req.:</td>
<td>Yes</td>
</tr>
<tr>
<td>Status:</td>
<td>Implemented</td>
</tr>
</tbody>
</table>

Change attribute values of all selected requirements at once.
ReqView References

• ReqView Homepage
  https://www.reqview.com

• ReqView Desktop Online
  https://www.reqview.com/apps/desktop/ReqViewDesktop.html

• ReqView Desktop at Chrome Store
  https://chrome.google.com/webstore/detail/reqview-desktop/cmbnhmeabefpflildnahpbghcejkbll

• Support Site - online help, knowledge base and discussion forum
  http://support.reqview.com/solution/categories