

# Entendendo o Components json

O `components.json` é um arquivo que possui uma estrutura do tipo JSON e usado pelo [editor de views](#) para exibir informações e configurações de suas propriedades, eventos e responsabilidades de cada componente (Figura 1). É através dele que se consegue adicionar mais recursos a um componente, permitindo sua customização sem a necessidade de configurá-lo via código.

Dentre alguns exemplos da necessidade do `components.json` no Cronapp, estão:

- Alteração do nome e ícone exibido no componente usando o editor de *views*;
- Personalização de elementos e atributos no *template* de forma visual;
- Internacionalização do controle;
- Inclusão de eventos a elementos do controle e outros.

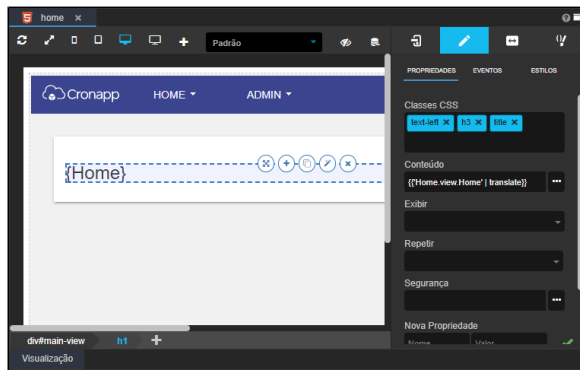


Figura 1 - Editor de *views* mostrando as propriedades do componente selecionado

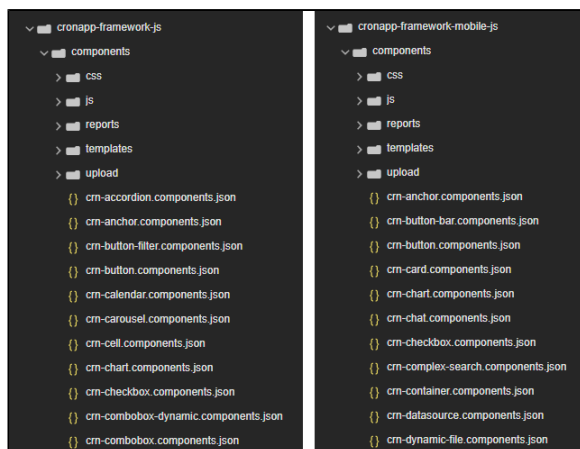
## Estrutura do arquivo

A estrutura do `components.json` é baseada em [JSON](#) (*JavaScript Object Notation* - Notação de Objetos JavaScript) que possui uma formatação simples e de fácil leitura e escrita, tanto para computadores quanto para humanos. Foi criado para facilitar a comunicação entre sistemas, independente da linguagem usada.

Os componentes padrões do Cronapp possuem seus próprios arquivos que terminam com a extensão `components.json` e seus diretórios mudam conforme o tipo de componente - ou seja, se o tipo é *web* ou *mobile* (Figura 2.1).

Os arquivos `components.json` só serão mostrados quando o [modo avançado](#) estiver habilitado.

- Componente *mobile*: `src/main/mobileapp/www/node_modules/cronapp-framework-mobile-js/components/`
- Componente *web*: `src/main/webapp/node_modules/cronapp-framework-js/components/`



## Nesta página

- [Estrutura do arquivo](#)
- [Propriedades do arquivo](#)
  - [Principais propriedades](#)
  - [Outras propriedades](#)
  - [attributesForPreview](#)
    - [values](#)
  - [childrenProperties](#)
  - [handleRules](#)
    - [rules](#)
  - [properties](#)
    - [Características](#)

## Conteúdo complementar

- [Estilos - CSS](#)
- [Eventos dos componentes visuais](#)
- [Propriedades dos componentes visuais](#)
- [Formatação de máscaras na camada servidor](#)
- [Formatação de máscaras na camada cliente](#)

Figura 2.1 - Arquivos JSON de cada componente padrão do Cronapp, sendo *web* à direita e *mobile* à esquerda

O Cronapp permite [criar componentes visuais](#) e para cada novo componente, são criados os arquivos `*.template.html` e `*.components.json`. Ambos arquivos devem ficar no mesmo diretório (destaque na Figura 2.2).

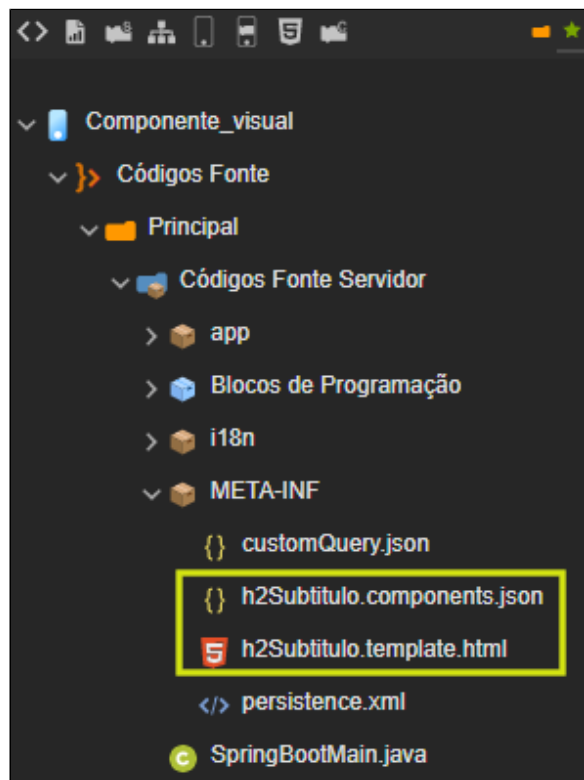


Figura 2.2 - Arquivos gerados ao criar componente visual

## Propriedades do arquivo

As propriedades nesses arquivos mudam conforme o componente.

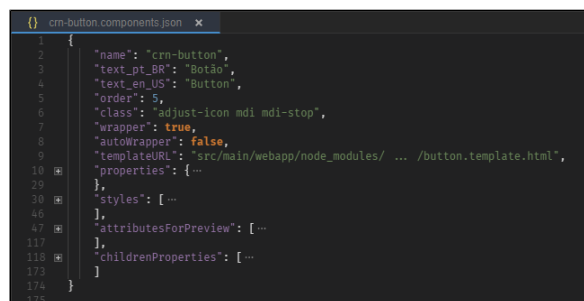


Figura 3 - Arquivo do componente visual Botão

## Principais propriedades

São aquelas encontradas na maioria dos componentes.

Propriedade	Objetivo	Possíveis valores
<b>name</b>	Atributo id do componente HTML.	<b>Ex.:</b> "name": "crn-button",
<b>text_pt_BR</b>	Nome do componente que aparecerá na aba controle do editor de <i>views</i> quando a IDE estiver no idioma português (Brasil).	<b>Ex.:</b> "text_pt_BR": "Botão",
<b>text_en_US</b>	Nome do componente que aparecerá na aba controle do editor de <i>views</i> quando a IDE estiver no idioma inglês (EUA).	<b>Ex.:</b> "text_en_US": "Button",
<b>class</b>	Define o ícone do controle que aparecerá no editor de <i>views</i> .  <b>OBS.:</b> Diferente da aplicação que permite o uso dos ícones <a href="#">Glyphicons</a> , <a href="#">font awesome icon</a> e <a href="#">Material Design Icons</a> , os botões dos componentes no editor de <i>views</i> permite apenas o <a href="#">Material Design Icons</a> .	<b>Ex.:</b> "class": "adjust-icon mdi mdi-stop",
<b>templateURL</b>	Endereço do template do componente na IDE.	<b>Ex.:</b> "templateURL": "src/main/webapp/node_modules/cronapp-framework-js/dist/components/templates/button.template.html",
<b>properties</b>	Lista as diretivas Angular, elementos HTML ou propriedades do Cronapp para permitir sua edição nas propriedades do componente no editor de <i>views</i> .	Consulte o tópico <a href="#">properties</a> .
<b>styles</b>	Lista os elementos HTML que compõem o template, possibilitando selecioná-los e personaliza-los no editor de <i>views</i> . Essa seleção pode ser feita utilizando as subabas dentro das abas <b>Propriedades</b> no editor de <i>views</i> ou selecionando o componente na tela, abrindo o ícone do cadeado e selecionando novamente os elementos internos do componente.	<p>Para cada elemento informado na lista:</p> <ul style="list-style-type: none"> <li>• <b>selector:</b> informa qual elemento HTML será selecionado. Se o template possuir dois elementos HTML iguais, é possível utilizar o <b>selector</b> do CSS para referenciá-lo;</li> <li>• <b>text_pt_BR:</b> internacionalização em português do Brasil para o rótulo da subaba que aparecerá em <b>propriedades</b> e <b>eventos</b> no editor de <i>views</i> referente ao elemento HTML selecionado;</li> <li>• <b>text_en_US:</b> internacionalização em inglês do EUA para o rótulo da subaba que aparecerá em <b>propriedades</b> e <b>eventos</b> no editor de <i>views</i> referente ao elemento HTML selecionado.</li> </ul> <p><b>Ex.:</b> "styles": [{   "selector": "div#{id} button",   "text_pt_BR": "Plano de fundo do botão",   "text_en_US": "Button background" }],</p>

<b>description</b>  <b>description_pt_BR</b>  <b>description_en_US</b>	<p>Define o <i>tooltip</i> que será exibido quando o cursor do mouse estiver posicionado sobre o botão do componente visual na paleta de controles do Editor de views.</p> <p>É possível definir um texto para cada uma das línguas disponíveis (português ou inglês) ou usar um valor genérico que será exibido independente da língua.</p>	<p>É possível utilizar uma opção ou todas.</p> <ul style="list-style-type: none"> <li>• <b>description</b>: valor padrão, será utilizado para definir o valor para o idioma português (pt_BR) e inglês (en_US), caso não possua;</li> <li>• <b>description_en_US</b>: valor usado quando a IDE estiver com o idioma em inglês;</li> <li>• <b>description_pt_BR</b>: valor usado quando a IDE estiver com o idioma em português. Essa opção é desnecessária ao utilizar a opção <b>description</b>.</li> </ul> <p><b>Ex. :</b>  "\"description\": \"Exibição de gráfico com valores da fonte de dados\",  \"description_en_US\": \"Graph display with data source values\",</p>
<b>category</b>  <b>category_pt_BR</b>  <b>category_en_US</b>	<p>Define quais categorias da paleta de controle serão exibidos o botão do componente visual.</p> <p>Não é obrigado que um componente esteja associado a uma categoria, mas caso isso ocorra, ele será listado apenas na categoria "Todos".</p> <p>Existem algumas categorias padronizadas do Cronapp (lista abaixo), porém é possível criar uma nova categoria apenas informando o novo nome na lista.</p> <p>Valores padrão:</p> <ol style="list-style-type: none"> <li>1. <b>INPUTS</b>: Controles de entrada de dados, <ul style="list-style-type: none"> <li>• en_US: Inputs;</li> <li>• pt_BR: Entradas</li> </ul> </li> <li>2. <b>BUTTONS</b>: Controles com botões, <ul style="list-style-type: none"> <li>• en_US: Buttons;</li> <li>• pt_BR: Botões</li> </ul> </li> <li>3. <b>PLUGINS</b>: Controles proveniente de plugins, <ul style="list-style-type: none"> <li>• en_US: Plugins</li> <li>• pt_BR: Plugins</li> </ul> </li> <li>4. <b>LAYOUTS</b>: Controles que constam layout. <ul style="list-style-type: none"> <li>• en_US: Layouts</li> <li>• pt_BR: Layouts</li> </ul> </li> <li>5. <b>FORMS</b>: Controles que formam formulários. <ul style="list-style-type: none"> <li>• en_US: Forms</li> <li>• pt_BR: Formulários</li> </ul> </li> <li>6. <b>GRIDS</b>: Grades, listas e Tabelas. <ul style="list-style-type: none"> <li>• en_US: Grids and Tables</li> <li>• pt_BR: Grades e Tabelas</li> </ul> </li> <li>7. <b>LISTS</b>: Controles com listas. <ul style="list-style-type: none"> <li>• en_US: Lists</li> <li>• pt_BR: Listas</li> </ul> </li> <li>8. <b>COMBOS</b>: Comboxes fixos e dinâmicos. <ul style="list-style-type: none"> <li>• en_US: Combos</li> <li>• pt_BR: Combos</li> </ul> </li> <li>9. <b>TEXTS</b>: Textos. <ul style="list-style-type: none"> <li>• en_US: Texts</li> <li>• pt_BR: Textos</li> </ul> </li> </ol>	<p>É possível utilizar uma opção ou todas.</p> <ul style="list-style-type: none"> <li>• <b>category</b>: lista padrão, será utilizado para informar a categoria nos idiomas português (pt_BR) e inglês (en_US), caso não possua;</li> <li>• <b>category_en_US</b>: lista usada quando a IDE estiver com o idioma em inglês;</li> <li>• <b>category_pt_BR</b>: lista usada quando a IDE estiver com o idioma em português. Essa opção é desnecessária ao utilizar a opção <b>description</b>.</li> </ul> <p><b>Ex. :</b>  "\"category\": [  \"BUTTONS\", \"FORMS\",  \"Minha nova categoria\" ],  \"category_en_US\":  [ \"BUTTONS\",  \"FORMS\", \"My new category\" ],</p>

## Outras propriedades

A utilização dessas propriedades varia de acordo com o componente.

Propriedade	Objetivo	Possíveis valores
<a href="#">attributesForPreview</a>	Lista de propriedades customizadas nos atributos CSS dos componentes.	Consulte o tópico <a href="#">attributesForPreview</a> .
<b>autoWrapper</b>	Adiciona uma tag <div> em torno do template do componente, facilitando a modificação da diagramação.	<ul style="list-style-type: none"> <li><b>true</b>: Permite tratamento diferenciado entre o componente e a &lt;div&gt;.</li> <li><b>false</b>: Não permite tratamento diferenciado entre o componente e a &lt;div&gt; e é o valor <i>default</i> quando a propriedade não é informada.</li> </ul> <b>Ex.:</b> "autoWrapper": false,
<a href="#">childrenProperties</a>	Lista as diretivas Angular, elementos HTML ou exclusivas do Cronapp para permitir sua edição nas propriedades dos subcomponentes (filhos) do componente no editor de <i>views</i> .	Consulte o tópico <a href="#">childrenProperties</a> .
<b>dependences</b>	Possui uma lista (editor) com os endereços de arquivos CSS específica para esse controle.	<b>Ex.:</b> "dependences": { "editor": [ "node_modules /cronapp-framework-js /components/css/multiselect- editor.css" ]},
<b>designTimeHTMLURL</b>	Sinaliza qual o caminho do html que será exibido no <i>designTime</i> .	<b>Ex.:</b> "designTimeHTMLURL": "src /main/webapp/node_modules /cronapp-framework-js/dist /components/templates/cron- scheduler.designtime.html",
<b>designTimeSelector</b>	Sinaliza qual diretiva ele irá usar.	<b>Ex.:</b> "designTimeSelector": "cron-scheduler",
<b>forcedGroup</b>	Força o agrupamento dos componentes HTML no template, permitindo a seleção de todos os subcomponentes de uma só vez.	<ul style="list-style-type: none"> <li><b>true</b>: Agrupa todos os subcomponentes, necessário realizar o desbloqueio para poder selecionar cada subcomponente.</li> <li><b>false</b>: Permite a seleção de cada subcomponente sem a necessidade do desbloqueio e é o valor <i>default</i> quando a propriedade não é informada.</li> </ul> <b>Ex.:</b> "forcedGroup": true,
<a href="#">handleRules</a>	Define características para algumas propriedades e regras para elementos do componente.	Consulte o tópico <a href="#">handleRules</a> .
<b>mask-placeholder</b>	Informa qual componente HTML receberá o texto do campo Sugestão no editor de <i>views</i> .	Possui em seu interior o valor "selector" onde será informado o componente HTML.  <b>Ex.:</b> "name": "mask-placeholder"
<b>onDrop</b>	Abre a janela de configuração especificada quando o componente é arrastado para a área de edição da view.	Nome do atributo.  <b>Ex.:</b> "onDrop": "openEditor",
<b>onDoubleClick</b>	Abre a janela de configuração especificada quando o componente recebe um duplo click na área de edição da view.	Nome do atributo.  <b>Ex.:</b> "onDoubleClick": "openEditor",
<b>pallette</b>	Oculto ou exibe o controle no editor de <i>views</i> .	<ul style="list-style-type: none"> <li><b>true</b>: exibe o controle no editor de <i>views</i> e é o valor <i>default</i> quando a propriedade não é informada.</li> <li><b>false</b>: oculta o controle no editor de <i>views</i>.</li> </ul> <b>Ex.:</b> "pallette": false,

<b>rows</b>	Define o tamanho do campo em número de linhas.	<b>Ex.:</b> "rows": { "displayName": "rows" },
<b>template</b>	Template HTML do componente. Por este ser um arquivo JSON, será necessário utilizar caracteres de escape para utilizar alguns caracteres.  <b>Exemplo:</b> "\n" e "\""	<b>Ex.:</b> <i class=\"fa fa-star\"/>
<b>text</b>	Nome do componente que aparecerá no editor de <i>views</i> quando não houver internacionalização (propriedades: text_en_US e text_pt_BR).	<b>Ex.:</b> "text": "File Upload Button",
<b>wrapper</b>	Adiciona uma tag <div> em torno do template do componente.	<ul style="list-style-type: none"> <li>• <b>true:</b> adiciona uma tag &lt;div&gt; e é o valor <i>default</i> quando a propriedade não é informada.</li> <li>• <b>false:</b> não adiciona uma tag &lt;div&gt;.</li> </ul> <b>Ex.:</b> "wrapper": false,

## attributesForPreview

Lista de propriedades customizadas para os atributos CSS dos componentes. Veja a tabela de opções dessa propriedade após o código de exemplo.

crn-text.components.json
<pre>"attributesForPreview": [   {     "name": "xattr-position",     "type": "btngroup",     "target": "class",     "values": [       {         "key": "text-left",         "value": "Standard",         "icon": "mdi mdi-ray-start"       }     ]   } ]</pre>

Propriedade	Objetivo	Possíveis valores
<b>name</b>	Nome da propriedade que terá suas características alteradas.	<b>Ex.:</b> "name": "xattr-theme",
<b>type</b>	Define a forma que será exibida as opções da propriedade. Normalmente são utilizados o grupo de botões ou seletor de itens.	<ul style="list-style-type: none"> <li>• <b>btngroup:</b> para exibir as propriedades em um grupo de botões alinhados. Permite a utilização de ícones nos botões.</li> <li>• <b>options:</b> para exibir as propriedades em um seletor de itens (<i>dropdown</i>). Permite a utilização de ícones, cores ou ambos.</li> </ul> <b>Ex.:</b> "type": "btngroup",
<b>target</b>	Informa o atributo ou elemento que será customizado. Normalmente utilizado para adicionar <b>class</b> ou <b>style</b> .	<b>Ex.:</b> "target": "class",
<b>values</b>	Lista das opções da propriedade.	Consulte o tópico <a href="#">values</a> .

## values

Lista das opções da propriedade **values** do [attributesForPreview](#).

Propriedade	Objetivo	Possíveis valores (exemplos)
<b>key</b>	Classe CSS que será adicionado no template após o usuário escolher o value.	<b>Ex.:</b> "key": "text-left",
<b>value</b>	Nome da opção que será exibida para o usuário.	<b>Ex.:</b> "value": "Standard",
<b>icon</b>	Define o ícone que será exibido ao lado do nome da opção.	<b>Ex.:</b> "icon": "mdi mdi-ray-start"
<b>color</b>	Exibe um quadrado com a cor especificada ao lado do nome da opção.  Observação: Utilizar apenas quando <b>type</b> é definido como <b>options</b> .	<b>Ex.:</b> "color": "#fff"

## childrenProperties

Responsável por criar propriedades distintas entre subcomponentes dentro de um componente. Veja a tabela de opções dessa propriedade após o código de exemplo.

Trecho de código do arquivo <code>crn-dynamic-menu.components.json</code>
<pre>"childrenProperties": [   {     "name": "options",     "selector": "cron-dynamic-menu",     "displayName_pt_BR": "Configuração",     "displayName_en_US": "Configuration",     "type": "text",     "order": 1,     "mandatory": true   } ]</pre>

Propriedade	Objetivo	Possíveis valores
<b>displayName_en_US</b>	Nome do campo que aparecerá na aba propriedades do editor de views quando a IDE estiver no idioma inglês (EUA).	<b>Ex.:</b> "displayName_en_US": "Title",
<b>displayName_pt_BR</b>	Nome do campo que aparecerá na aba propriedades do editor de views quando a IDE estiver no idioma português (BR).	<b>Ex.:</b> "displayName_pt_BR": "Título",
<b>editExpression</b>	Exibe o ícone <b>Editar expressão</b> ao lado do nome do campo e ao clicar é aberta uma caixa de texto para a edição da expressão ou inserção de Fonte de dados.	<ul style="list-style-type: none"><li><b>true</b>: ativa a opção;</li><li><b>false</b>: desativa a opção e é o valor <i>default</i> quando a propriedade não é informada.</li></ul> <b>Ex.:</b> "editExpression": false,
<b>mandatory</b>	Define uma propriedade como preenchimento obrigatório, adicionando contorno vermelho à propriedade quando vazia.	<ul style="list-style-type: none"><li><b>true</b>: define como obrigatória.</li><li><b>false</b>: define como não obrigatória.</li></ul> <b>Ex.:</b> "mandatory": true,

<b>name</b>	Informa o atributo do elemento HTML que será manipulado em um campo do editor de <i>views</i> .	<b>Ex.:</b> "name": "ng-model",
<b>onDisplay</b>	Executa comando em JavaScript para atualizar o componente na tela sempre que uma alteração for feita.	Com o comando javascript:function() { /* função JavaScript */ }  <b>Ex.:</b> "onDisplay": "javascript: function() { return arguments[0].replace('vars.',''); }"
<b>onSave</b>	Executa comando em JavaScript para salvar sempre que uma opção for selecionada ou ação tomada no componente.	Com o comando javascript:function() { /* função JavaScript */ }  <b>Ex.:</b> "onSave": "javascript: function() { var result = arguments [0]; if (result.indexOf('.') == -1 && result.trim().length > 0 ) { result = 'vars.'+result; } return result; }"
<b>selector</b>	Informa qual atributo do elemento HTML será selecionado para manipulação.	<b>Ex.:</b> "selector": "ui-select",
<b>type</b>	Define a forma de como o subcomponente irá se comportar como permitir apenas texto, seletor ou importação de fonte de dados.	<ul style="list-style-type: none"> <li>• <b>type:</b> campo seletor <i>input</i> (ex: <i>text</i>, <i>number</i>, <i>date</i>, <i>email</i>, <i>money</i>, <i>telephone</i>, <i>time</i>, <i>url</i>, etc);</li> <li>• <b>text:</b> texto normal;</li> <li>• <b>event:</b> direciona para a aba de "eventos";</li> <li>• <b>logic:</b> booleano (sim ou não);</li> <li>• <b>content:</b> direciona para a aba de "conteúdo";</li> <li>• <b>icon:</b> abre o modal de adicionar ícone;</li> <li>• <b>pageDatasource:</b> campo seletor de fonte de dados (para funcionar é necessário inserir uma fonte de dados);</li> <li>• <b>allDatasourceFields:</b> captura as variáveis da fonte de dados (para funcionar é necessário inserir uma fonte de dados);</li> <li>• <b>projectResource:</b> abre um modal que busca os arquivos da extensão sinalizado no "<i>resourceType</i>", como imagem.</li> </ul> <b>Ex.:</b> "type": "event",

## handleRules

Define características para algumas propriedades e regras para elementos do componente. Veja abaixo a tabela de opções dessa propriedade.

Trecho de código do arquivo <i>crn-grid.components.json</i>
<pre> "handleRules": {   "canOrder": true,   "canActivate": true,   "canIncrement": true,   "rules": [     {       "activeSelector": ".table-header th.active",       "activeClass": "active",       "sourceHTML": "&lt;th class=\"k-header\" data-container=\"true\" rowspan=\"\" colspan=\"\"&gt;&lt;div data-container=\"true\" data-replace=\" true\"&gt;New Title&lt;/div&gt;&lt;/th&gt;",       "targetSelector": ".table-header"     }   ] }</pre>



Propriedade	Objetivo	Possíveis valores / exemplos
<b>canActivate</b>	Ativa ou desativa as regras.	<ul style="list-style-type: none"> <li><b>true</b>: ativa o <code>handleRules</code> do componente.</li> <li><b>false</b>: desativa o <code>handleRules</code> do componente.</li> </ul> <b>Ex. :</b> <code>"canActivate": true,</code>
<b>canIncrement</b>	Exibe o ícone <b>adicionar</b> ao lado do nome do campo e ao clicar é inserido outro item na lista.  Exemplo: outra coluna em uma tabela ou outro item em uma lista de opções.	<ul style="list-style-type: none"> <li><b>true</b>: insere o ícone <b>adicionar</b>;</li> <li><b>false</b>: oculta o ícone <b>adicionar</b> e é o valor <i>default</i> quando a propriedade não é informada.</li> </ul> <b>Ex. :</b> <code>"canIncrement": true,</code>
<b>canOrder</b>	Define a possibilidade de ordenação dos itens.	<ul style="list-style-type: none"> <li><b>true</b>: possibilita ordenação dos itens e é o valor <i>default</i> a propriedade não é informada;</li> <li><b>false</b>: impede a ordenação dos itens.</li> </ul> <b>Ex. :</b> <code>"canOrder": true,</code>
<b>canToggle</b>	Mostra o ícone <b>exibir</b> ao lado do nome do campo e esse ícone recolhe ou expande os itens ou submenus desse controle no editor de views, permitindo visualizar e selecionar seu conteúdo para personalização.  Exemplo: caixa de seleção ou menu.	<ul style="list-style-type: none"> <li><b>true</b>: insere o ícone <b>exibir</b>;</li> <li><b>false</b>: oculta o ícone <b>exibir</b> e é o valor <i>default</i> a propriedade não é informada.</li> </ul> <b>Ex. :</b> <code>"canToggle": true,</code>
<b>rules</b>	Define algumas regras para alteração do template do componente a partir de determinadas ações.	Consulte o tópico <a href="#">rules</a> .

## rules

Lista de as regras de controle do [handleRules](#).

Propriedade	Objetivo	Possíveis valores / exemplos
<b>accordionTabClicked</b>	Específico do componente acordeão, usado para acionar a ação de expandir ou recolher seus itens.	<b>Ex. :</b> <code>"accordionTabClicked": "panel-collapse.collapse",</code>
<b>activeClass</b>	Ativa uma classe específica apontada anteriormente.	<b>Ex. :</b> <code>"activeClass": "in",</code>
<b>activeSelector</b>	Seleciona o atributo que sofrerá a alteração.	<b>Ex. :</b> <code>"activeSelector": "div.panel-collapse.collapse.in",</code>

<b>sourceHTML</b>	Define trecho do HTML que será acrescentado ou retirado do <i>template</i> , caso a regra seja atendida.	<b>Ex.:</b> "sourceHTML": "<li></li>",
<b>targetSelector</b>	Seleciona um elemento HTML ou classe CSS para sofrer a alteração,	<b>Ex.:</b> "targetSelector": "div.panel-group",
<b>toggleClass</b>	Adicionar ou retirar uma classe CSS.	<b>Ex.:</b> "toggleClass": "open",
<b>toggleSelector</b>	Adicionar ou retirar um elemento seletor.	<b>Ex.:</b> "toggleSelector": ".dropdown",

## properties

Responsável por criar propriedades e eventos que afetam todo o componente (incluindo os subcomponentes). Veja abaixo a tabela de opções dessa propriedade.

### Trecho de código do arquivo `crn-title.components.json`

```
"properties": {
  "xattr-fullsize": {
    "displayName_en_US": "Width",
    "displayName_pt_BR": "Largura"
  },
  "xattr-theme": {
    "displayName_en_US": "Theme",
    "displayName_pt_BR": "Tema"
  },
  "disabled": {
    "displayName_en_US": "Disabled",
    "displayName_pt_BR": "Desabilitado"
  },
  "class": {
    "order": 9999
  },
  "id": {
    "order": 5
  }
},
```

Propriedade	Campo no editor de views	Objetivo
<b>auto-refresh</b>	Atualizar Automaticamente (ms)	Define o tempo, em milissegundos, que o datasource atualizará automaticamente.
<b>class</b>	Classes CSS	Adiciona classes CSS já criadas.
<b>checkrequired</b>	Validar Campos Requeridos	Insere ou atualiza os dados somente se todos os campos marcados como requeridos estiverem preenchidos.
<b>crn-datasource</b>	Fonte de dados	Define a fonte de dados do componente.
<b>cronapp-filter-autopost</b>	Postagem Automática	Realiza consulta ou ação a cada letra digitada no campo.
<b>cronapp-filter-caseinsensitive</b>	Case Insensitive	Define se o campo diferenciará letras minúsculas e maiúsculas.

<b>cronapp-filter-operator</b>	Operador	Define operadores relacionais e de igualdade. (=, <, >, <= e >=)
<b>dependent-by</b>	Depende de	Cria uma dependência do componente para a exibição de conteúdo.  <b>Ex.:</b> As informações sobre um usuário só irão aparecer na tela quando um dos usuários for selecionado.
<b>dependent-lazy-post</b>	Dependente de postagem sob demanda	Cria uma dependência do componente com a fonte de dados informada para a inserção de conteúdo. Nesse campo deve ser informado o objeto independente em relação ao campo "dependente de postagem de campo sob demanda".  <b>Ex.:</b> As regras para um usuário só serão submetidas se existir um usuário.
<b>dependent-lazy-post-field</b>	Dependente de postagem de campo sob demanda	Cria uma dependência do componente com a fonte de dados informada para a inserção de conteúdo. Nesse campo deve ser informado o objeto ou id que ficará dependente.  <b>Ex.:</b> As regras para um usuário só serão submetidas se existir um usuário.
<b>entity</b>	Origem de dados	Define qual a fonte de dados que irá alimentar o componente.
<b>href</b>	Referência	Define uma URL para gerar um <i>link</i> .
<b>id</b>	Identificador	Atributo que especifica um ID exclusivo para o componente. Por padrão um id é gerado automaticamente e atribuído ao componente.
<b>keys</b>	Chaves	Define quais parâmetros serão usados como chave na fonte de dados.
<b>lazy</b>	Postergar Carga	Alimenta a fonte de dados somente após uma ação do usuário.
<b>load-data-strategy</b>	Estratégia para Carregar Dados	Define as condições de carregamento dos dados
<b>mask</b>	Máscara	Lista pré-definida de máscaras.  <b>Ex.:</b> CEP, CPF, data, telefone etc.
<b>mask-placeholder</b>	Sugestão de máscara	Possibilita adição de novas máscaras customizadas.
<b>ng-click</b>	Ao clicar	Executa evento ao clicar (AngularJS).
<b>ng-dblclick</b>	Ao duplo clicar	Executa evento ao duplo clicar (AngularJS).
<b>ng-init</b>	Ao iniciar	Executa evento ao iniciar (AngularJS).
<b>ng-model</b>	Valor	Propriedade AngularJS que pode ser usada para obter e alimentar o conteúdo pelo bloco de programação.
<b>ng-mouseenter</b>	Ao entrar com o mouse	Executa evento ao entrar com o mouse (AngularJS).
<b>ng-mouseleave</b>	Ao sair com o mouse	Executa evento ao sair com o mouse (AngularJS).
<b>ng-mousemove</b>	Ao mover mouse	Executa evento ao mover mouse (AngularJS).
<b>on-after-create</b>	Após criar	Executa um evento após a criação algo.
<b>on-after-delete</b>	on-after-delete	Executa evento após ação de deletar algo.
<b>on-after-fill</b>	Após preencher	Executa evento após preenchimento de um campo.

<b>on-after-update</b>	Após atualizar	Executa evento após atualização de algo.
<b>on-before-create</b>	Antes de criar	Executa evento antes de criar algo.
<b>on-before-delete</b>	Antes de deletar	Executa evento antes de deletar algo.
<b>on-before-update</b>	Antes de atualizar	Executa evento antes de atualizar algo.
<b>onclick</b>	Ao clicar	Executa evento ao clicar no componente (JavaScript).
<b>on-error</b>	Ao errar	Executa um evento quando ocorrer um erro no carregamento de algum arquivo ou mídia (JavaScript).
<b>onmouseover</b>	Ao clicar	Executa evento no precionar do clique do botão (JavaScript).
<b>parameters</b>	Mestre Detalhe	Vincula um ou mais atributos de uma fonte de dados com outra fonte de dados.
<b>parameters-null-strategy</b>	Estratégia de Cláusula Nula	Define as condições de carregamento dos dados.
<b>qr</b>	QR Code	Exclusivo para o controle QR Code.
<b>src</b>	Origem	Define campo para carregamento de imagem.
<b>target</b>	Alvo	Seleciona em qual janela será aberta o link.
<b>xattr-fullsize</b>	Largura	Determina se o componente ocupará a linha inteira ou não. Suas opções padrões são <b>inline</b> ou <b>block</b> e definidas no <a href="#">attributesForPreview</a> .
<b>xattr-position</b>	Posição	Determina em qual posição o componente ficará alinhado. Suas opções padrões são à esquerda ( <i>standard</i> ), centro ( <i>middle</i> ) ou direita ( <i>right</i> ) e definidas no <a href="#">attributesForPreview</a> .
<b>xattr-size</b>	Tamanho	Determina o tamanho do componente na tela. Suas opções padrões são <i>standard</i> , <i>larger</i> (btn-lg) e <i>small</i> (btn-sm) e definidas no <a href="#">attributesForPreview</a> .
<b>xattr-theme</b>	Tema	Determina o tema do componente. Suas opções padrões são <i>default</i> , <i>primary</i> , <i>success</i> , <i>info</i> , <i>warning</i> , <i>danger</i> e <i>link</i> e definidas no <a href="#">attributesForPreview</a> .
<b>xattr-type</b>	Style Type	Determina o estilo do botão. Suas opções padrões são <i>default</i> , <i>clear</i> (button-clear) e <i>outline</i> (button-outline) e definidas no <a href="#">attributesForPreview</a> .
<b>disabled</b>	Desativado	Determina se o botão será desabilitado ou não.

Acesse a documentação [Eventos dos componentes visuais](#) para mais detalhes sobre cada evento.

## Características

Define as características das propriedades e evento do controle de qualquer uma das propriedades informadas na tabela de [properties](#).

Propriedade	Objetivo	Possíveis valores
<b>displayName_en_US</b>	Nome do rótulo que aparecerá na aba propriedades do editor de views quando a IDE estiver no idioma inglês (EUA).	<b>Ex.:</b> "displayName_en_US": "Configuration",
<b>displayName_pt_BR</b>	Nome do rótulo que aparecerá na aba propriedades do editor de views quando a IDE estiver no idioma português (BRA).	<b>Ex.:</b> "displayName_pt_BR": "Configurações",

<b>mandatory</b>	Define uma propriedade como preenchimento obrigatório, adicionando contorno vermelho à propriedade quando vazia.	<ul style="list-style-type: none"> <li>• <b>true</b>: define como obrigatória;</li> <li>• <b>false</b>: define como não obrigatória.</li> </ul> <b>Ex.:</b> "mandatory": true,
<b>options</b>	Define as chaves e os valores (tanto em inglês quanto em português) que serão exibidas quando a propriedade for do tipo lista.	<b>Ex.:</b> "options": [{ "key": "default", "value_en_US": "Default", "value_pt_BR": "Padrão" }],
<b>removable</b>	Exibe o ícone remover propriedade ao lado do nome do campo e ao clicar, o campo será removido dentre as propriedades ou eventos.  <b>OBS.:</b> É possível chamar novamente a propriedade através do campo <b>nova propriedade</b> .	<ul style="list-style-type: none"> <li>• <b>true</b>: insere o ícone;</li> <li>• <b>false</b>: oculta o ícone e é o valor <i>default</i> a propriedade não é informada.</li> </ul> <b>Ex.:</b> "removable": true,
<b>resourceType</b>	Define qual a extensão de arquivo será utilizada.	Permite imagens do tipo jpeg, gif ou png.  <b>Ex.:</b> "resourceType": "image/jpeg,image/gif,image/png,image/svg+xml",