

Eventos dos componentes visuais

Um evento normalmente espera uma interação do usuário para realizar uma ação programada, por exemplo: ao clicar em um botão, uma janela será aberta; ao mover o mouse sobre um componente, as coordenadas X,Y do ponteiro do mouse são exibidas ou ao iniciar a tela, uma mensagem de boas vindas é exibida para o usuário.

Alguns eventos são específicos para determinados componentes, outros são genéricos e podem ser usados em qualquer componente. Nessa página veremos como configurar uma ação para um evento.

Seleção da ação

Para selecionar a ação de um dos eventos, clique no botão "... " para abrir a janela seleção da ação (Figura 1). Ela possui 5 abas:

1. **Expressão:** permite a inserir uma ação manualmente;
2. **Bloco de programação:** seleciona (ou cria) uma função de bloco de programação;
3. **Funções:** permite selecionar um bloco de programação sem a necessidade de criar uma função de bloco de programação;
4. **Relatório:** abre um relatório (exclusivo para projetos web);
5. **Formulário:** abre uma página web ou mobile do projeto.

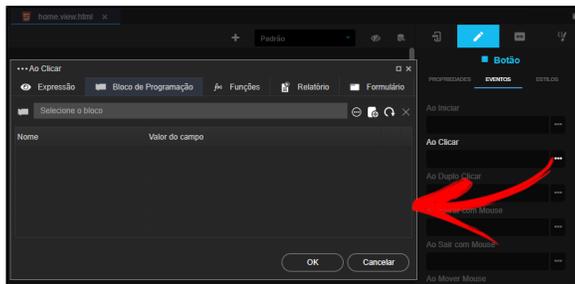


Figura 1 - Janela de seleção de ação para o evento "Ao clicar" do componente botão

Valor do campo

Blocos de Programação, Funções, Relatórios e Formulários permitem passagem de parâmetros em suas chamadas. Dessa forma, ao selecionar uma dessas abas, podemos escolher passar valores estáticos ou expressões (dinâmicos), como no exemplo da figura 2.

A caixa de seleção **Valor do campo** definirá se o conteúdo passado será estático ou dinâmico e a depender da opção selecionada, um componente específico é exibido. As opções para o **Valor do campo** são:

- (estático) **Texto:** exibe um campo de texto;
- (estático) **Numérico:** exibe um campo que só permite valores numéricos;
- (estático) **Data e Hora:** exibe campos para inserção de data e hora;
- (estático) **Hora:** exibe campo para inserção de hora;
- (estático) **Lógico:** exibe campo para definir valor verdadeiro ou falso;
- (dinâmico) **Expressão:** exibe uma caixa de seleção com diversos elementos da tela atual, dentre esses elementos temos:
 - propriedade `ng-model` (Valor) dos componentes,
 - fontes de dados e campos das fontes de dados,
 - objeto do evento,
 - quando a janela é aberta a partir das configurações de um componente (exemplo grade ou lista avançada), elementos desse componente também serão exibidos.

Nesta página

- **Seleção da ação**
 - Valor do campo
 - Aba Expressão
 - Aba Bloco de programação
 - Aba Funções
 - Aba Relatório
 - Aba Formulário
- **Eventos do mouse**
 - Ao clicar (On Click)
 - Ao duplo clicar (On Double Click)
 - Ao entrar com o mouse (On Mouse Enter)
 - Ao sair com o mouse (On Mouse Leave)
 - Ao mover mouse (On Mouse Move)
- **Eventos do componente**
 - Ao iniciar (nginit)
 - Ao alterar (On change)
 - Ao focar (On focus)
 - Ao mover slider (On slide)
 - Ao sair do foco (On blur)
- **Eventos de clipboard**
 - Ao copiar (On copy)
 - Ao colar (On paste)
- **Eventos de teclado**
 - Apertar tecla (On keydown)
 - Ao soltar tecla (On keyup)
- **Eventos da fonte de dados**
 - Ao mudar de status (On change status)
 - Antes de atualizar (On before update)
 - Antes de criar (On before create)
 - Antes de deletar (On before delete)
 - Ao errar (On error)
 - Após deletar (After delete)
 - Após atualizar (On after update)
 - Após criar (On after create)
 - Após preencher (On after fill)
 - Para atualizar dados (To update data)
 - Para inserir dados (To insert data)
 - Para obter dados (To get data)
 - Para remover dados (To remove data)
- **Eventos da grade**
 - Antes de editar (Before edit)
 - Ao cancelar (On cancel)
 - Ao editar (On edit)
 - Ao ligar os dados (On data binding)
 - Ao remover (On remove)
 - Ao salvar (On save)

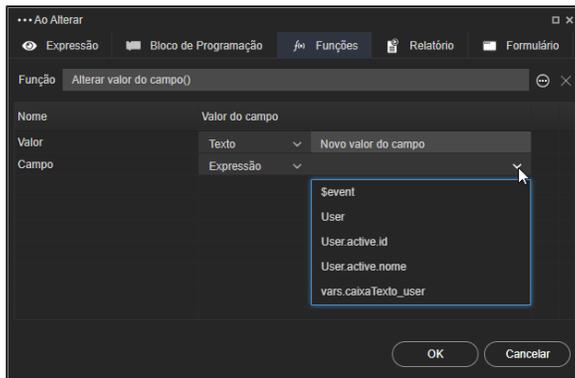


Figura 2 - Seleção de um valor dinâmico da tela para o atributo Campo

Para adicionar dados que não estão listados na caixa de seleção quando o **Valor do campo** está apontado como *Expressão*, como por exemplo o identificador de um elemento HTML, basta selecionar a opção "Texto" em **Valor do campo** e inserir o conteúdo manualmente na caixa de texto ao lado.

Aba Expressão

Possui um campo de texto para adicionar comandos manualmente (high-code), aceita internacionalização (item 2 da figura 3). Ao selecionar uma ação em uma das outras abas, o comando resultante será exibido nessa área. No título da janela (item 1 da Figura 3) é exibido o evento que chamará essa ação,

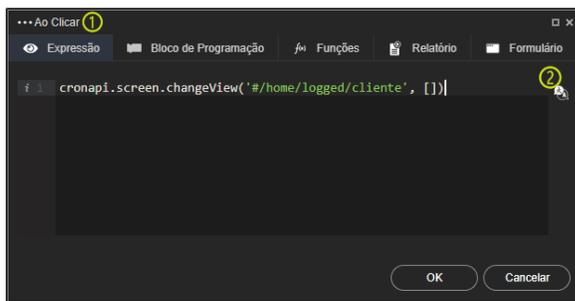


Figura 3 - Aba Expressão da janela de seleção da ação

Aba Bloco de programação

Permite selecionar uma função de bloco de programação (cliente ou servidor) criado no projeto como ação de um evento.

Aviso

 Lembramos que o uso de bloco do tipo servidor pode resultar em falhas de segurança.

- Ao salvar alterações (On save changes)
- Ao vincular dados (On data bound)
- **Eventos de caixa de seleção**
 - Ao carregar (On data bound)
 - Ao fechar (On close)
 - Ao filtrar (On filtering)
 - Ao selecionar (On select)
 - Ao remover seleção ()
- **Eventos do editor de texto rico**
 - Antes de inserir conteúdo (On before content)
 - Antes de processar (Pre process)
 - Ao alterar (On change)
 - Ao clicar (On click)
 - Ao colar (On paste)
 - Ao colar antes do processamento (On paste pre process)
 - Ao colar depois do processamento (On paste post process)
 - Ao desfazer
 - Ao duplo clicar (On double click)
 - Ao entrar com mouse (On mouse enter)
 - Ao executar comando (On exec command)
 - Ao focar (On Focus)
 - Ao iniciar (On start)
 - Ao inserir conteúdo (On set content)
 - Ao mover mouse (On mouse move)
 - Ao mudar de nó (On node change)
 - Ao obter conteúdo (On get content)
 - Ao refazer (On redo)
 - Ao sair com mouse (On mouse leave)
 - Ao sair do foco (On blur)
 - Ao soltar tecla (On keyup)
 - Ao sujar editor (On dirty)
 - Apertar tecla (On keydown)
 - Após processar (Post process)
- **Eventos mobile**
 - Ao clique rápido (On tap)
 - Ao clique duplo rápido (On double tap)
 - Ao pressionar (On hold)
 - Ao arrastar (On drag)
 - Ao deslizar (On swipe)
 - Ao deslizar para esquerda (On swipe left)
 - Ao deslizar para a direita (On swipe right)

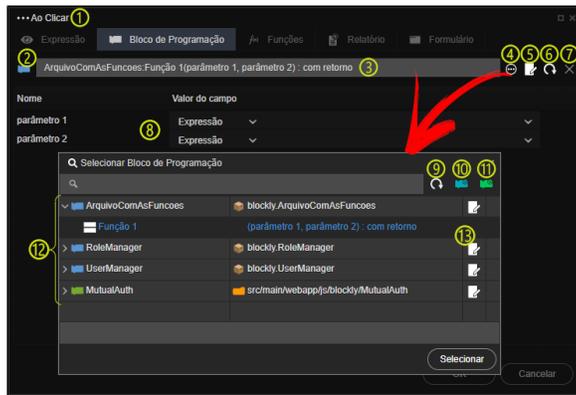


Figura 4 - Seleção de uma função na aba Bloco de Programação

1. **Evento:** informa o evento que chamará essa ação;
2. **Ícone da função:** sua cor informa a função selecionada:
 - Azul: função servidor;
 - Verde: função cliente;
3. **Função selecionada:** informações da função selecionada, possui a seguinte sintaxe:

```
<Arquivo>:<Função>([ <Parâmetro 1> [ , <Parâmetro n> ] ] ) : <Retorno>
```

 - **Arquivo:** informa o nome do arquivo Java ou JavaScript onde está a função selecionada;
 - **Função:** informa o nome da função selecionada;
 - **Parâmetro:** exibe os parâmetros que são recebidos pela função;
 - **Retorno:** informa se a função possui retorno;
4. **Seleção do bloco de programação:** Abre a janela que lista os blocos de programação existentes;
5. **Novo arquivo:** permite criar arquivo blockly enquanto não existir função selecionada;
6. **Atualizar:** verifica e atualiza alterações na função selecionada;
7. **Limpar:** limpa a seleção feita;
8. **Lista de parâmetros:** exibe todos os parâmetros recebidos pela função selecionada, possui 3 colunas:
 - **Nome:** nome do parâmetro;
 - **Valor do campo:** exibe os tipos do parâmetro
 - valores estáticos: Texto, Numérico, Data e hora, Hora, Lógico;
 - valores dinâmicos: Expressão;
 - **Conteúdo:** conteúdo a ser inserido com base no valor do campo;
9. **Atualizar:** atualiza a lista de funções;
10. **Novo arquivo Servidor:** mesma função do item 5, porém este só permite criar arquivo blockly do tipo servidor;
11. **Novo arquivo Cliente:** mesma função do item 5, porém este só permite criar arquivo blockly do tipo cliente;
12. **Lista de funções:** exibe os arquivos de bloco de programação, ao expandir um arquivo é possível visualizar todas as funções, seus parâmetros e se possui retorno;
13. **Editor blockly:** abre uma janela para edição do arquivo de bloco selecionado.

Aba Funções

Quando usamos a aba Bloco de Programação, precisamos criar uma função dentro do arquivo blockly para depois chamá-la na ação do evento, fundamental para funções mais complexas. Porém, ações que necessitem de um único bloco de programação, como [Exibir componente](#), [Abrir url](#), [Alterar valor do campo](#) e outros, podem ser chamados diretamente pela aba Funções, não sendo necessário criar uma função blockly para chamar esse bloco.

A aba Funções só lista os blocos do tipo cliente e são exatamente os mesmos blocos das funções de bloco de programação, ou seja, possuem os mesmos nomes e parâmetros. Ao selecionar um bloco, seus atributos são listados abaixo (item 8 da figura 5) para definir o que será passado, podendo ser um [valor estático](#) ou [expressão](#).

- Componentes visuais
 - Estilos - CSS
 - Eventos dos componentes visuais
 - Propriedades dos componentes visuais
 - Formatação de máscaras na camada servidor
 - Formatação de máscaras na camada cliente
- Internacionalização
- Responsividade
- Customização de templates

Assista sobre o tema no Cronapp Academy

Caso seja seu primeiro acesso ao Cronapp Academy, crie antes uma conta gratuita e matricule-se no curso abaixo.

- Aula: [Funções em botões](#)

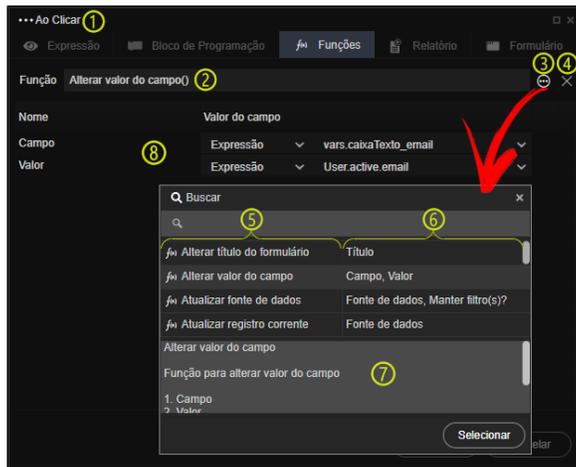


Figura 5 - A aba Funções é usada para ações onde é necessário apenas 1 bloco de programação

Aba Relatório

Utilize a aba relatório para, ao executar a ação, abrir o relatório selecionado no componente [Visualizar de relatório](#) dentro de um modal na tela. Não existe a opção de abrir relatórios em projetos mobile, sendo uma funcionalidade exclusiva para projetos web.

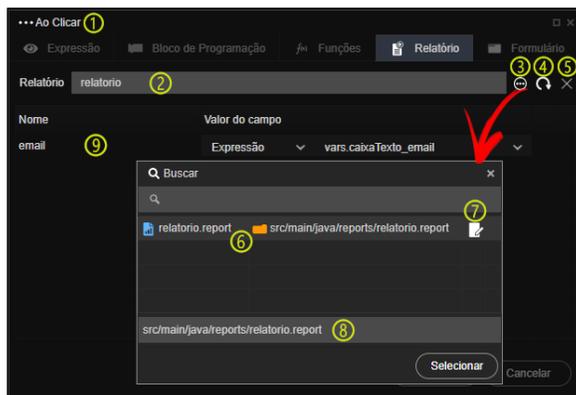


Figura 5 - Aba Relatório da janela de seleção da ação

1. **Evento:** informa o evento que chamará essa ação;
2. **Relatório:** exibe o nome do relatório selecionado;
3. **Seleção do relatório:** abre a janela que lista os relatórios existentes;
4. **Atualizar:** verifica e atualiza alterações no relatório selecionado;
5. **Limpar:** limpa a seleção feita;
6. **Lista de relatório:** exibe a lista de relatório, informando o nome e o local onde estão dentro do projeto;
7. **Editor relatório:** abre uma janela para edição do relatório selecionado;
8. **Relatório selecionado:** informa qual relatório está atualmente selecionado;
9. **Lista de parâmetros:** exibe todos os parâmetros recebidos pelo relatório selecionado, possui 3 colunas:
 - **Nome:** nome do parâmetro;
 - **Valor do campo:** exibe os tipos do parâmetro
 - **valores estáticos:** Texto, Numérico, Data e hora, Hora, Lógico;
 - **valores dinâmicos:** Expressão;
 - **Conteúdo:** conteúdo a ser inserido com base no valor do campo;

Aba Formulário

Permite acessar outras páginas e formulários do projeto, inserindo parâmetros de tela e adicionando filtros à fontes de dados. Os parâmetros de tela e filtros são passados via [query string](#).

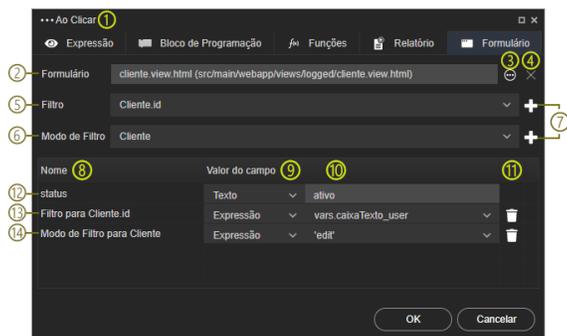
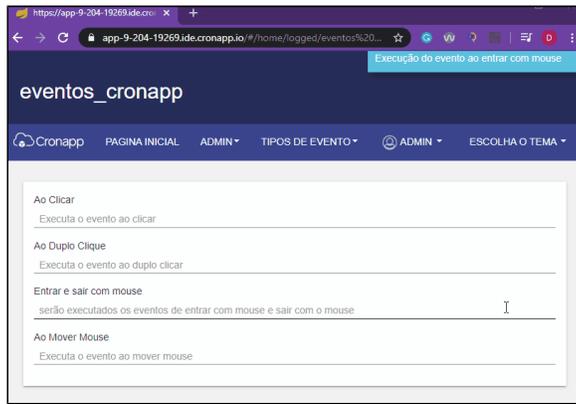


Figura 6 - Aba Formulário da janela de seleção da ação

1. **Evento:** informa o evento que chamará essa ação;
2. **Formulário:** exibe o nome do formulário view selecionado;
3. **Seleção do formulário:** abre a janela que lista todas as páginas e formulários existentes no projeto;
4. **Limpar:** limpa a seleção feita;
5. **Filtro:** lista todos campos das fontes de dados existentes no formulário selecionado;
6. **Modo de Filtro:** lista todas as fontes de dados existente no formulário selecionado;
7. **"+":** adiciona o filtro ou modo de filtro na lista de parâmetros;
8. **Coluna nome:** exibe o nome do parâmetro de tela ou dos filtros e modos de filtro adicionados. Ao adicionar um filtro, este aparecerá com o texto "Filtro para " na frente, o mesmo ocorrerá ao adicionar um modo de filtro;
9. **Valor do campo:** exibe os tipos do parâmetro:
 - **valores estáticos:** Texto, Numérico, Data e hora, Hora, Lógico;
 - **valores dinâmicos:** Expressão;
10. **Conteúdo:** conteúdo a ser inserido com base no valor do campo;
11. **Apagar:** exclui o filtro ou modo de filtro selecionado, não é possível apagar os parâmetros de tela de uma página;
12. **Parâmetro de tela:** quando a página selecionada possuir **parâmetros de tela** configurado, estes aparecerão na lista de atributos. No exemplo da figura 6, o formulário "cliente.view.html" possui o parâmetro de tela "status" e estamos passando o valor estático "ativo";
13. **Filtro:** ao adicionar um filtro na chamada do formulário, a fonte de dados selecionada restringirá sua consulta sem a necessidade de qualquer configuração na fonte de dados;
14. **Modo de Filtro:** permite abrir o formulário em diferentes modos, ao selecionar o valor de campo como "Expressão" serão exibidas as seguintes opções:
 - **normal:** abre o formulário listando todos os registros da fonte de dados, é possível restringir o conteúdo adicionando filtro;
 - **edit:** abre o formulário exibindo os campos em modo edição de registros da fonte de dados, sendo necessário adicionar no filtro o item que será editado;
 - **insert:** abre o formulário exibindo os campos em modo inserção de registro, nesse caso, não devemos inserir filtros.

Eventos do mouse

Os eventos a seguir executam alguma ação a partir de ações do mouse, como o movimentar do cursor do mouse sobre um componente, duplo clique, clique único etc.



Alguns eventos de mouse sendo executados

Ao clicar (On Click)

ng-click

Executa uma ação quando o componente receber um clique com o mouse ou um toque na tela.

Ao duplo clicar (On Double Click)

ng-dblclick

Executa uma ação quando o usuário clica duas vezes seguidas com o botão do mouse em um componente.

Ao entrar com o mouse (On Mouse Enter)

ng-mouseenter

Executa uma ação quando o ponteiro do mouse é movido de fora para dentro do elemento.

Ao sair com o mouse (On Mouse Leave)

ng-mouseleave

Executa uma ação quando o ponteiro do mouse é movido de dentro para fora do elemento.

Ao mover mouse (On Mouse Move)

ng-mousemove

Executa uma ação enquanto o ponteiro do mouse está sendo movido sobre o elemento.

Eventos do componente

Alguns componentes podem trabalhar com o mesmo evento, enquanto outros não. Isso parte da funcionalidade tanto do componente quanto de como ocorre o evento.



Execução do evento *ao iniciar*

Ao iniciar (nginit)

ng-init

Executa uma ação assim que o componente é renderizado na tela.

Ao alterar (On change)

ng-change

Executa uma ação assim que algo no componente é alterado.

Componentes: entrada de texto, entrada de texto com botão, entrada de texto com ícone, grade, caixa de seleção fixa, caixa de seleção dinâmica, caixa de seleção múltipla, auto completar, auto completar dinâmica e slider.

Ao focar (On focus)

ng-focus

Executa uma ação assim que o componente é focado.

Ao mover slider (On slide)

ng-slide

Executa uma ação sempre que o slider é arrastado na barra.

Componente: slider.

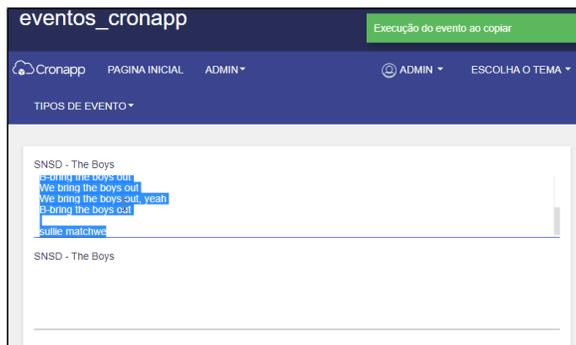
Ao sair do foco (On blur)

ng-blur

Executa uma ação assim que o componente sai do foco.

Eventos de clipboard

Os eventos a seguir executam ações de copiar, colar, recortar, etc.



Eventos de clipboard (copiar e colar)

Ao copiar (On copy)

ng-copy

Executa uma ação assim que o elemento é copiado do componente.

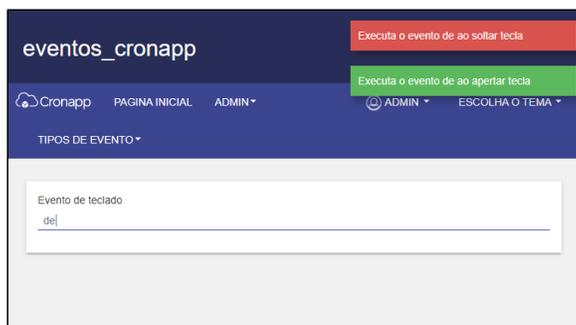
Ao colar (On paste)

ng-paste

Executa uma ação assim que o elemento é colado no componente.

Eventos de teclado

Os eventos a seguir executam alguma ação a partir de ações do teclado, como apertar ou soltar uma tecla.



Eventos de apertar e soltar uma tecla

Apertar tecla (On keydown)

ng-keydown

Executa uma ação assim que uma tecla é associada no teclado.

Ao soltar tecla (On keyup)

ng-keyup

Executa uma ação ao soltar uma tecla do teclado.

Eventos da fonte de dados

Os eventos a seguir executam ações a partir do componente **fonte de dados**.

ANIME	EPISODIO	STATUS
Boku no Hero	82	Em andamento
Naruto	700	Finalizado
One Piece	928	Em andamento
Card Captor Sakura	70	Finalizado
Nagi no Asukara	26	Finalizado

Execução dos eventos antes de atualizar e após atualizar

Ao mudar de status (On change status)

on-change-status

Executa uma ação após a mudança de status (ex.: inserção ou edição).

Antes de atualizar (On before update)

on-before-update

Executa uma ação antes de um dado ser atualizado.

Antes de criar (On before create)

on-before-create

Executa uma ação antes de um novo dado ser criado.

Antes de deletar (On before delete)

on-before-delete

Executa uma ação antes de um novo dado ser excluído.

Ao errar (On error)

on-error

Executa uma ação quando um erro ocorre na fonte de dados.

Após deletar (After delete)

on-after-delete

Executa uma ação após um dado ser excluído.

Após atualizar (On after update)

on-after-update

Executa uma ação após um dado ser atualizado.

Após criar (On after create)

on-after-create

Executa uma ação após um dado ser criado.

Após preencher (On after fill)

on-after-fill

Executa uma ação após os dados serem carregados ou atualizados na tela.

Para atualizar dados (To update data)

on-put

Bloqueia a requisição de atualizar dados para executar a função do bloco.

Para inserir dados (To insert data)

on-post

Bloqueia a requisição de inserir dados para executar a função do bloco.

Para obter dados (To get data)

on-get

Bloqueia a requisição de obter dados para executar a função do bloco.

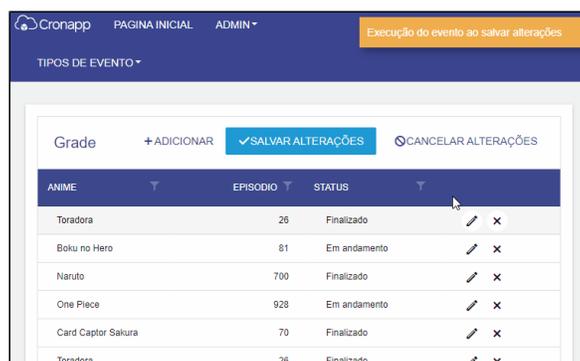
Para remover dados (To remove data)

on-delete

Bloqueia a requisição de remover dados para executar a função do bloco.

Eventos da grade

Os eventos a seguir executam ações a partir do componente **grade**.



Execução do evento ao salvar alterações

TIPOS DE EVENTO

Grade + ADICIONAR ✓ SALVAR ALTERAÇÕES CANCELAR ALTERAÇÕES

ANIME	EPISODIO	STATUS		
Toradora	26	Finalizado	✎	✕
Boku no Hero	81	Em andamento	✎	✕
Naruto	700	Finalizado	✎	✕
One Piece	928	Em andamento	✎	✕
Card Captor Sakura	70	Finalizado	✎	✕
Toradora	26	Finalizado	✎	✕

Execução dos eventos ao salvar e ao salvar alterações da grade

Antes de editar (Before edit)

ng-before-edit

Executa uma ação ao editar um dado na grade.

Ao cancelar (On cancel)

ng-cancel

Executa uma ação ao clicar no botão default de cancelar da grade.

Ao editar (On edit)

ng-edit

Executa uma ação ao clicar no botão default de editar da grade.

Ao ligar os dados (On data binding)

ng-data-binding

Executa uma ação quando for vincular os dados.

Ao remover (On remove)

ng-remove

Executa uma ação ao clicar no botão default de excluir da grade.

Ao salvar (On save)

ng-save

Executa uma ação ao clicar no botão default de salvar da grade.

Ao salvar alterações (On save changes)

ng-save-changes

Executa uma ação ao clicar no botão default de salvar alterações da grade.

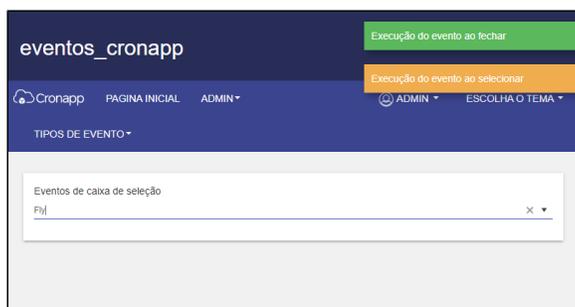
Ao vincular dados (On data bound)

ng-data-bound

Executa uma ação após ter os dados vinculados.

Eventos de caixa de seleção

Os eventos a seguir executam ações a partir de todos os componentes do tipo **caixa de seleção**, como caixa de seleção dinâmica e auto completar.



Execução dos eventos ao selecionar e ao fechar

Ao carregar (On data bound)

ng-data-bound

Executa uma ação enquanto dados são carregados na caixa de seleção ou no auto completar.

Ao fechar (On close)

ng-close

Executa uma ação assim que fecha as opções mostradas na caixa de seleção ou no auto completar.

Ao filtrar (On filtering)

ng-filtering

Executa uma ação assim que começa a digitar no campo da caixa de seleção ou de auto completar.

Ao selecionar (On select)

ng-select

Executa uma ação assim que selecionar a opção da caixa ou do auto completar.

Ao remover seleção ()

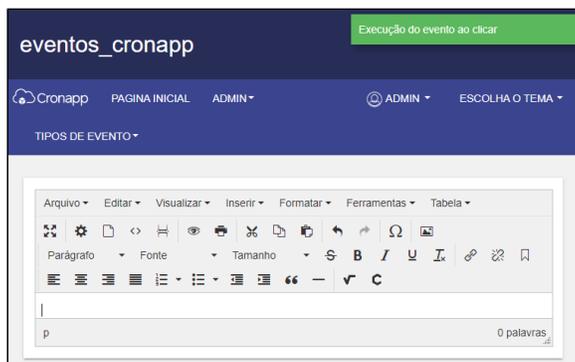
ng-deselect

Executa uma ação assim que um item é removido da caixa de seleção múltipla.

Componente: caixa de seleção múltipla.

Eventos do editor de texto rico

Os eventos a seguir executam ações a partir do componente **editor de texto rico**.



Execução do evento ao clicar no componente texto rico

Antes de inserir conteúdo (On before content)

ng-before-set-content

Executa uma ação antes que qualquer conteúdo possa ser inserido no componente.

Antes de processar (Pre process)

ng-pre-process

Executa uma ação quando for realizar alguma ação na área de edição do editor de texto rico, como digitar ou utilizar as funcionalidades do componente.

Ao alterar (On change)

ng-cron-change

Executa uma ação enquanto estiver ocorrendo alterações no componente, como digitar no campo ou selecionar um item.

Ao clicar (On click)

ng-cron-click

Executa uma ação quando o mouse clica no campo de digitação do componente.

Ao colar (On paste)

ng-cron-paste

Executa uma ação sempre que um conteúdo é colado no componente.

Ao colar antes do processamento (On paste pre process)

ng-paste-pre-process

Executa uma ação quando for colar um conteúdo na área de edição do componente.

Ao colar depois do processamento (On paste post process)

ng-paste-post-process

Executa uma ação após colar um conteúdo na área de edição do componente.

Ao desfazer

ng-cron-undo

Executa uma ação sempre que realiza uma ação de desfazer, através do botão na barra de ferramenta ou das teclas de atalho, no componente.

Ao duplo clicar (On double click)

ng-cron-dblclick

Executa uma ação assim que o campo de digitação do componente for clicado duas vezes.

Ao entrar com mouse (On mouse enter)

ng-cron-mouseenter

Executa uma ação quando o ponteiro do mouse é movido de fora para dentro do componente.

Ao executar comando (On exec command)

ng-exec-command

Executa uma ação assim que uma funcionalidade do editor é executada na área de edição, como adicionar parágrafo ou inserir tabela.

Ao focar (On Focus)

ng-cron-focus

Executa uma ação no momento que o componente é focado pelo usuário.

Ao iniciar (On start)

ng-cron-init

Executa uma ação assim que o componente for renderizado na tela.

Ao inserir conteúdo (On set content)

ng-set-content

Executa uma ação sempre que um conteúdo for inserido no componente.

Ao mover mouse (On mouse move)

ng-cron-mousemove

Executa uma ação enquanto o ponteiro do mouse está sendo movido sobre o elemento.

Ao mudar de nó (On node change)

ng-node-change

Executa uma ação sempre que uma mudança ocorre no componente.

Ao obter conteúdo (On get content)

ng-get-content

Executa uma ação todas as vezes que o editor recebe um conteúdo.

Ao refazer (On redo)

ng-cron-redo

Executa uma ação sempre que realiza uma ação de refazer, através do botão na barra de ferramenta ou das teclas de atalho, no componente.

Ao sair com mouse (On mouse leave)

ng-cron-mouseleave

Executa uma ação quando o ponteiro do mouse é movido de dentro para fora do elemento.

Ao sair do foco (On blur)

ng-cron-blur

Executa uma ação assim que o mouse clica em outro elemento da página quando o componente **texto rico** estava selecionado previamente.

Ao soltar tecla (On keyup)

ng-cron-keyup

Executa uma ação sempre que uma tecla é desapertada no componente.

Ao sujar editor (On dirty)

ng-cron-dirty

Executa uma ação sempre que algo é digitado no editor ou algo é selecionado na barra de ferramenta do editor, como torna o texto em negrito ou diminuir recuo.

Apertar tecla (On keydown)

ng-cron-keydown

Executa uma ação sempre que uma tecla é pressionada no componente.

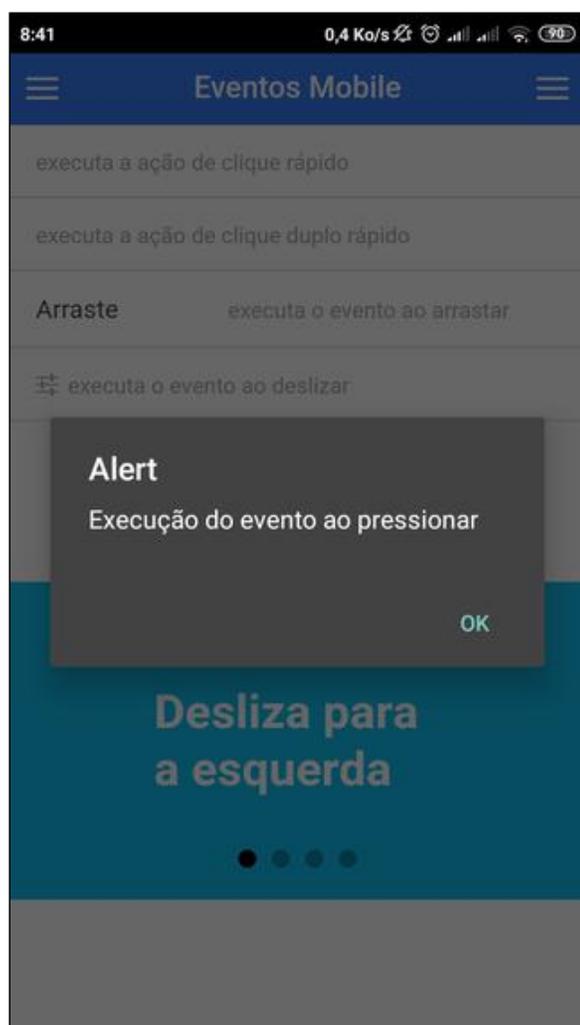
Após processar (Post process)

ng-post-process

Executa uma ação após realizar alguma ação na área de edição do editor de texto rico, como digitar ou utilizar as funcionalidades do componente.

Eventos mobile

Os eventos a seguir executam ações ocorrem somente em componentes mobile.



Execução de um evento no mobile

Ao clique rápido (On tap)

on-tap

Executa uma ação sempre que um elemento é clicado.

Ao clique duplo rápido (On double tap)

on-double-tap

Executa uma ação sempre que um elemento é clicado duas vezes.

Ao pressionar (On hold)

on-hold

Executa uma ação sempre que um elemento é pressionado sem soltar por alguns milissegundos.

Ao arrastar (On drag)

on-drag

Executa uma ação sempre que um elemento é arrastado.

Ao deslizar (On swipe)

on-swipe

Executa uma ação sempre que é feito uma ação de swipe no elemento.

Ao deslizar para esquerda (On swipe left)

on-swipe-left

Executa uma ação sempre que é feito uma ação de swipe no elemento para o lado esquerdo.

Componentes: carrossel e lista com swipe.

Ao deslizar para a direita (On swipe right)

on-swipe-right

Executa uma ação sempre que é feito uma ação de swipe no elemento para o lado direito.

Componentes: carrossel e lista com swipe.