Blocks and Events

Exercício – Como Fazer

# 

**Esboço**

**[Como](#_t71y8lcg117j) Fazer**

Block [StarDisplay](#_66aimibi03wg)

Mostrar Avaliação do Usuário e Média

Avaliação do Usuário

Avaliação Média

# 

# 

# Esboço

Nesse exercício iremos introduzir aos Blocks em nossa aplicação. Isso irá permitir aos usuários finais que avaliem um filme. Essa avaliação será baseada num sistema de avaliação por estrelas, onde o usuário poderá avaliar um filme de 1 a 5 estrelas. Essa funcionalidade de avaliação deve ser adicionada a tela MovieDetail.

Também na mesma tela, devemos mostrar a informação da média de avaliação de um filme, que irá combinar todas as avaliações dadas pelos usuários em um filme em particular.

Para completar isso, iremos

1. Criar um Block para mostrar o número de estrelas.
2. Nos assegurar que a avaliação será visível para qualquer usuário pelo número de estrelas preenchidas
3. Mostrar a avaliação média do filme.
4. Sem implementar as estrelas uma segunda vez, mostre o conjunto de estrelas para o usuário para que possa avaliar o filme.
5. Fazer as estrelas clicáveis e disparar a lógica para criar/atualizar a avaliação desse usuário para esse filme.
6. Garantir que essa avaliação média de estrelas não seja clicável.
7. Adaptar a aplicação para permitir que somente usuários registrados possam avaliar um filme.

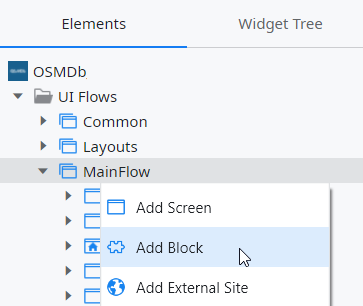
Um usuário pode avaliar o mesmo filme mais de uma vez. Nesses casos, a aplicação deve considerar como uma atualização da avaliação anterior e não como uma nova avaliação desse usuário.

# Como Fazer

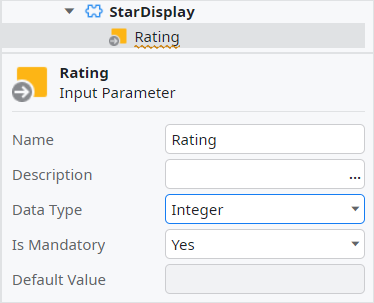
Nessa seção, vamos descrever o passo a passo do exercício *7 – Blocks e Eventos*.

## Block StarDisplay

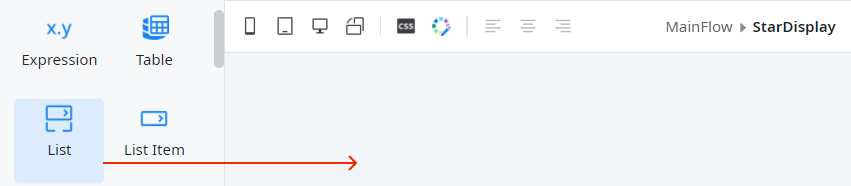
1. Crie um Block *StarDisplay* para mostrar uma List de estrelas. Isso vai depender de um parâmetro de entrada de Rating
   1. No módulo OSMDd, na aba Interface, clique como o botão direito em MainFlow e selecione **Add Block**

****

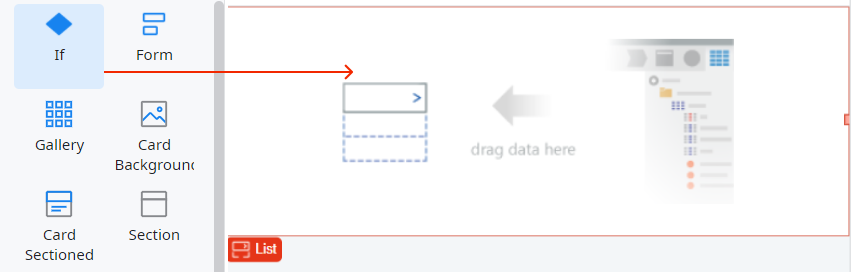
* 1. Defina o **Name** como do Block para *StarDisplay*
  2. Assim como nas telas, clique com o botão direito no Block e selecione **Add Input Parameter** para adicionar uma nova entrada. Defina seu **Name** para Rating e mude Data Type para Integer.



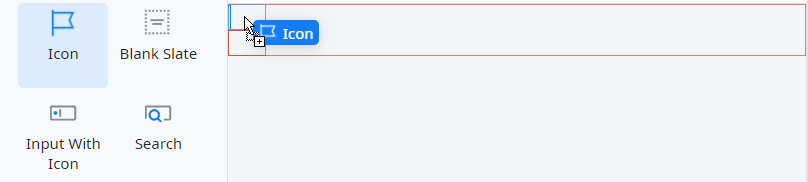
1. Nesse Block **StarDisplay**, crie uma List de estrelas que possam ser preenchidas com alguma cor ou vazias, dependendo da avaliação do filme. Por exemplo, se o filme tem uma avaliação de 3, teremos 3 estrelas coloridas e duas estrelas vazias.
   1. Arraste uma List ao a área do conteúdo principal do Block.



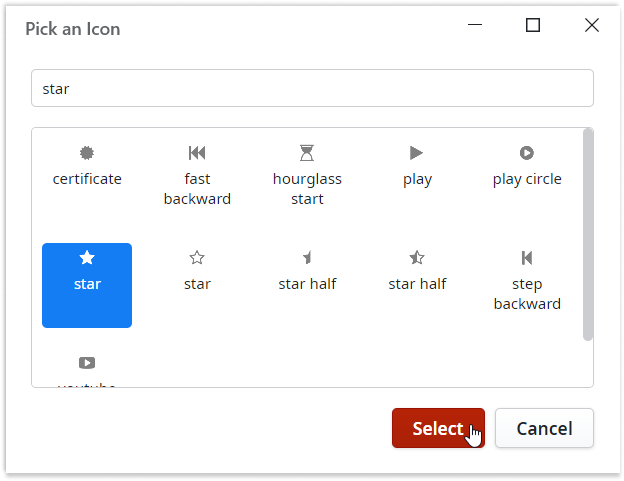
* 1. Arraste um If e solte dentro da List.



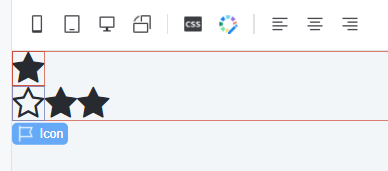
* 1. Arraste um Icon dentro do ramo True do If.



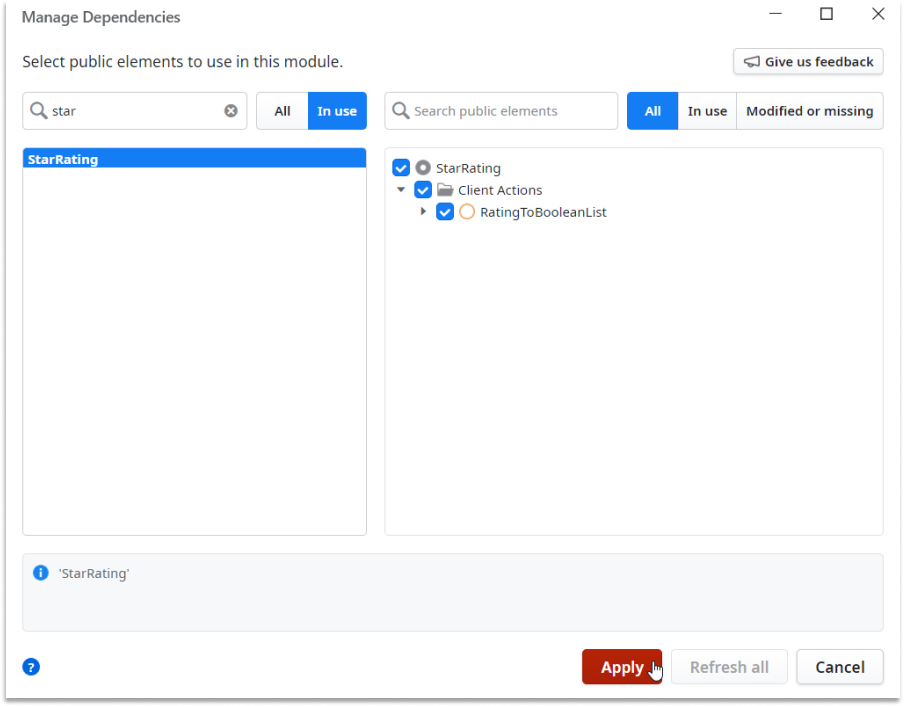
* 1. Na nova janela, selecione o ícone Star.



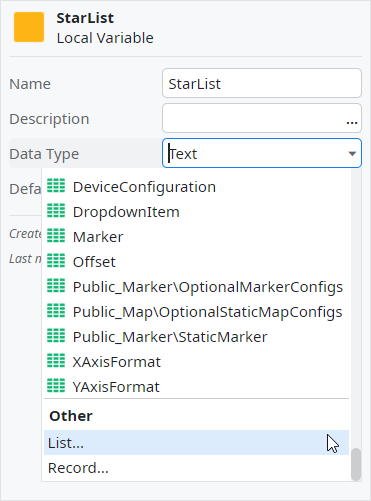
* 1. Nas propriedades do ícone defina o **Size** para 2x Font size
  2. Repita o mesmo processo para o ramo **False**, mas desta vez escolha o ícone **white star**. O Block deverá ter a aparência abaixo:



1. Agora que temos o UI no nosso Block, precisamos uma source de dados. Para isso, vamos usar a Action definida no módulo **StarRating**, chamado **RatingToBooleanList**. Essa Action pega a avaliação e transforma numa List de booleanos. Assim, quando o valor da List for True, a estrela colorida deverá aparecer. Por outro lado, a estrela branca deverá aparecer.
   1. Abra a janela Manage Dependencies, selecione o módulo **StarRating** e referencie a Action **RatingToBooleanList**.

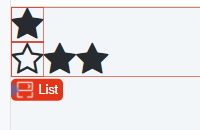
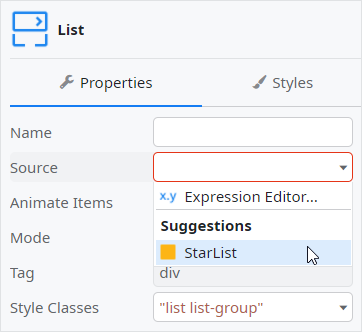


* 1. Adicione uma variável Local ao Block StarDisplay chamado *StarList*.Defina **Data Type** para *Boolean List*.



**NOTA:** Essa variável irá representar essa List booleana que virá da Action RatingToBooleanList e irá influenciar quantas estrelas preenchidas aparecerá.

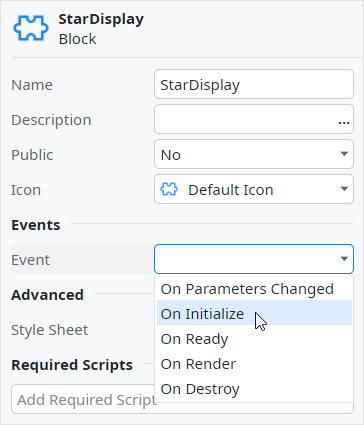
* 1. Selecione a List no Block e defina sua source para ser *StarList*.

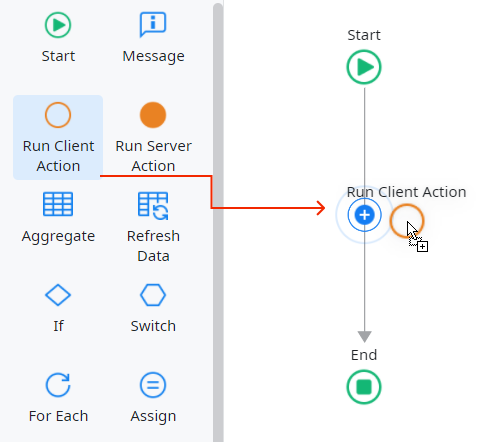
* 1. Selecione o If dentro da List e defina sua **Condition** para *StarList.Current*

**NOTA:** Como a variável StarList é uma List de booleanos, isso significa que cada elemento da List será true ou false. Se for true, o Block irá mostrar a estrela preenchida. Em OutSystems a List tem uma propriedade Current contendo o primeiro elemento como padrão. Contudo, quando usada numa List ou num loop, o Current pode ser usado como um iterador. Por exemplo, na List no Block, a primeira estrela usará o primeiro elemento da List, a segunda estrela usará o segundo elemento etc. Isso é feito automaticamente. O desenvolvedor simplesmente usa a propriedade Current da List.

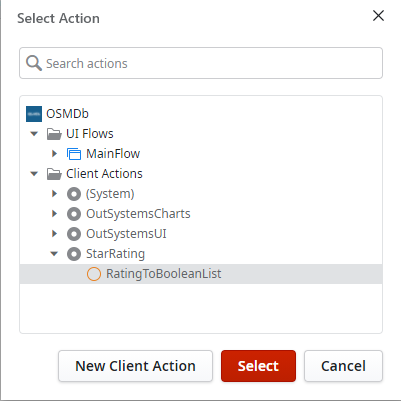
* 1. Como a variável StarList está vazia, precisamos inicializá-la com dados. Vamos usar a Action **RatingToBooleanList**. A questão é: Onde usar isso? Telas e Blocks possuem os mesmos Events que podem ser alavancados. Um deles é o evento **On Initialize**, que é disparado automaticamente quando a tela/Block é inicializada
  2. Nas propriedades do Block StarDisplay, em Events. Selecione a opção **On Initialize**.



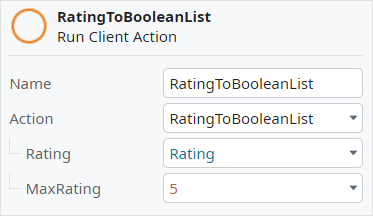
* 1. Isso cria uma nova Action no Block, chamada *OnInitialize*. Arraste e solte uma **Run Client Action** no novo fluxo.



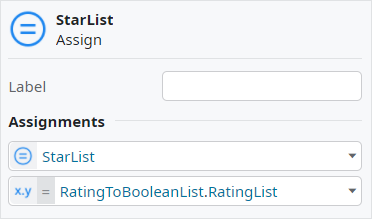
* 1. Na nova janela, selecione a Action **RatingToBooleanList**.



* 1. Defina a entrada **Rating** da Action para o parâmetro de entrada do Block Rating e o **MaxRating** para *5*.



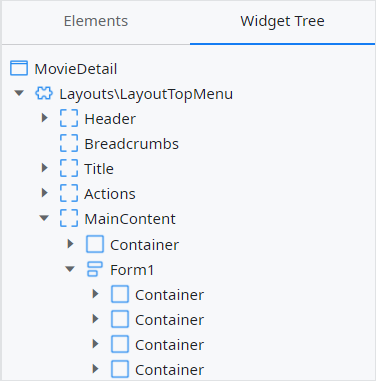
* 1. Arraste um Assign e solte depois das Action RatingToBooleanList. Defina a seguinte assignment para alimentar nossa variável local com a saída da Action.



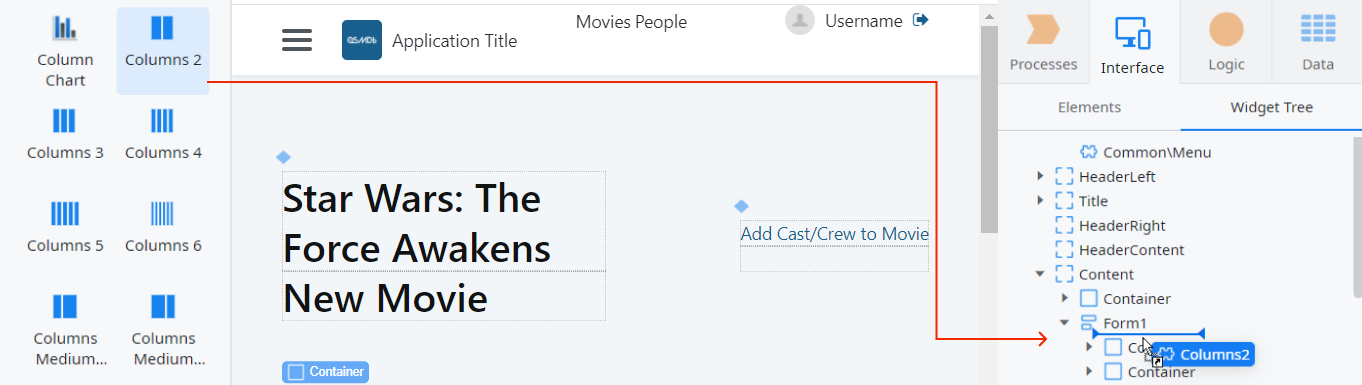
## Exibir o User e Average Rating

Na seção anterior, criamos a UI para essas estrelas no Block StarDisplay. Agora, vamos usar essa UI para mostrar as avaliações dos usuários e média na tela MovieDetail.

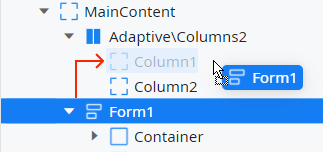
1. Vamos criar um espaço para as avaliações dos usuários e a média na tela MovieDetail. Divida a tela em duas colunas e use a da direita para mostrar as duas avaliações.
   1. Na tela MovieDetail, abra a Widget Tree e expanda a **MainContent**



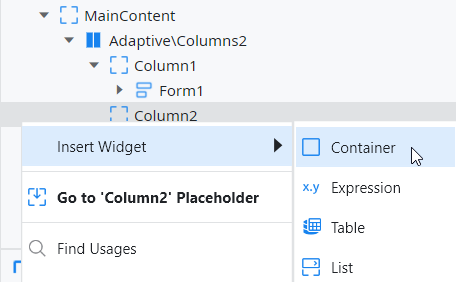
* 1. Arraste o widget **Columns 2** para o MainContent acima do **Form1**.



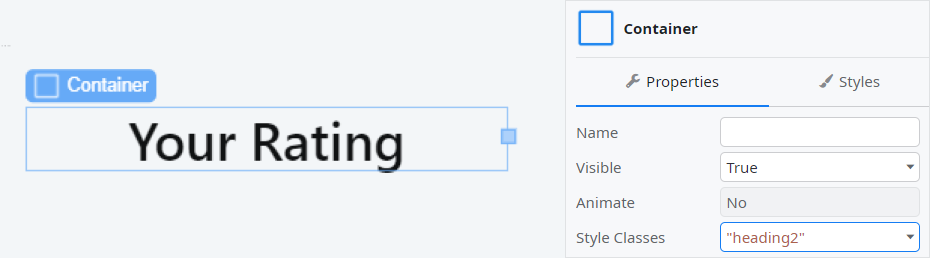
* 1. Expanda a Columns2 na Widget Tree, arraste o formulário e solte na **Column1**.



* 1. Adicione dois novos Containers na **Column2**

****

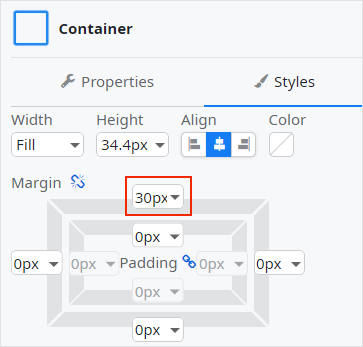
* 1. No primeiro Container, insira *Your Rating* e defina sua propriedade **Style Classes** para *“heading2”*.



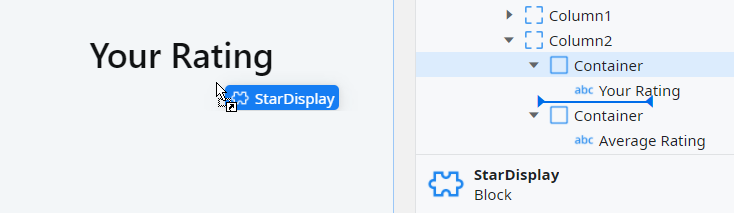
* 1. Centralize o texto dentro do Container



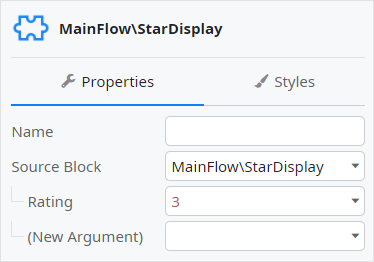
* 1. No segundo Container, insira *Average Rating* **Style Classes** para *“heading2”*. Centralize o texto dentro do Container também.
  2. Selecione o segundo Container, troque para a aba **Styles** e adicione a **Margin top** para 30px.



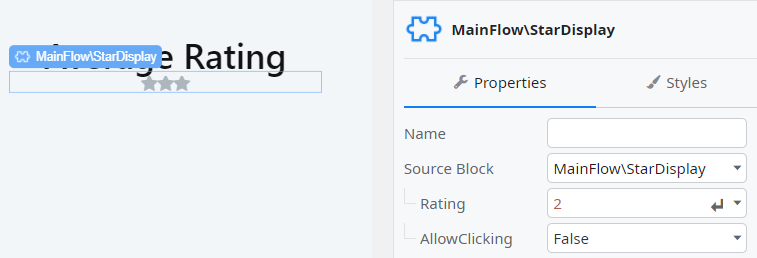
1. Use o Block StarDisplay para mostrar a avaliação do usuário e a média. Use valores estáticos apenas para ver se está funcionando.
   1. Arraste o Block **StarDisplay** e solte no primeiro Container, bem abaixo do texto Your Rating.



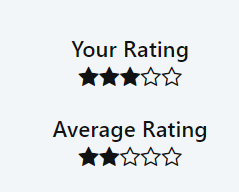
* 1. Defina a entrada do **Rating** do Block para 3.



* 1. Arraste novamente o Block **StarDisplay**, mas desta vez solte no segundo container, abaixo do texto Average Rating. Defina o **Rating** para *2*.



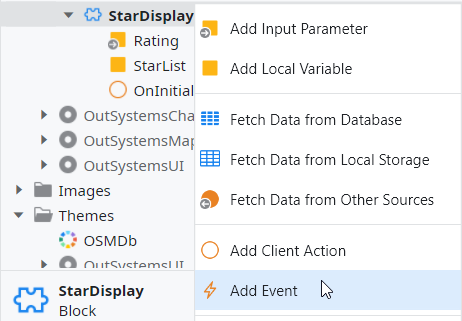
* 1. Publish o módulo e se assegure que as estrelas aparecem adequadamente no navegador.



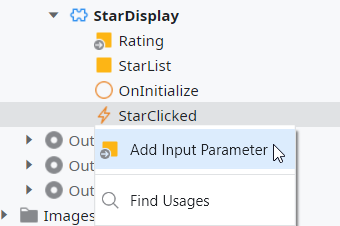
## User Rating

A tela MovieDetail já mostra as estrelas usando o Block StarDisplay. Agora, precisamos permitir aos usuários finais que avaliem o filme ao clicar nas estrelas. Assim, temos que ter certeza de que somente a avaliação do usuário seja clicável.

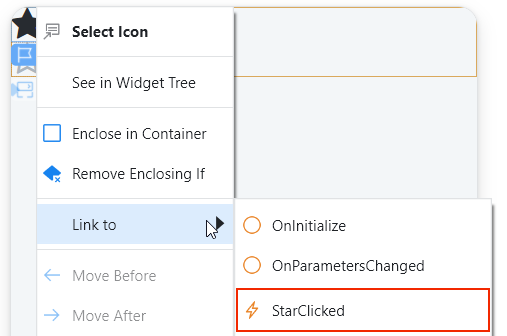
1. No Block **StarDisplay**, use os Links nas estrelas para suportar a funcionalidade de clicar. O link deve dar o gatilho de um evento que vai mandar informação da estrela clicada para o parent, que irá afetar também a avaliação.
   1. Abra o Block **StarDisplay**.
   2. Clique com o botão direito no Block StarDisplay e selecione **Add Event**.



* 1. Defina o **Name** do evento para *StarClicked*. Defina sua propriedade **Is Mandatory** para **No**.
  2. Adicione um parâmetro de entrada no Event e nomeie como *SelectedStar*, como o **Data Type** definido como *Integer*.

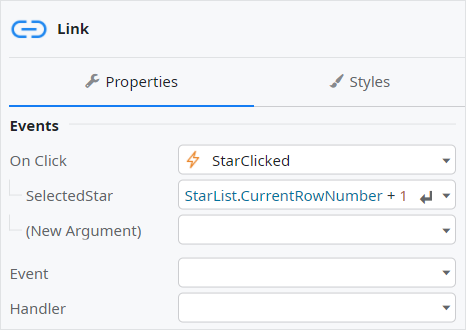


* 1. Abra o Block StarDisplay e clique com o botão direito no ícone da estrela preenchida. Selecione **Link to -> StarClicked**



**NOTA:** Podemos definir um Link para disparar um evento. Nesse caso, o evento será disparado quando o usuário final clicar no link.

* 1. Nas propriedades do novo Link, defina o valor da entrada do **SelectedStar para** *StarList.CurrentRowNumber + 1*

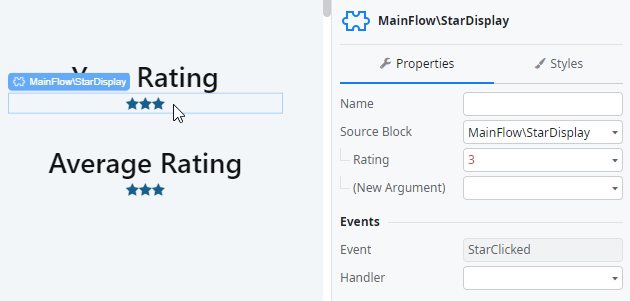
**

**NOTA:** A StarList é a variável local que é usada como source da List de estrelas no Block. Para saber qual estrela o usuário clicou, podemos usar a propriedade CurrentRowNumber que todas as Lists têm. O CurrentRowNumber tem a posição na List da propriedade Current (no qual tem o valor). Essa propriedade começa em zero, ao invés de um, consequentemente há a necessidade do somar um em nosso caso de uso.

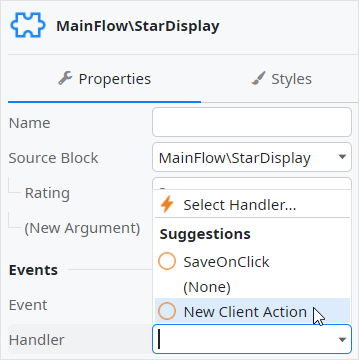
Vamos ver um exemplo. Numa List com {True, True, False, False}, o primeiro elemento tem valor True (Current = True) e a CurrentRowNumber será 0, já que estamos na primeira posição da List. O terceiro elemento seria Current = False e CurrentRowNumber = 2. Voltando ao nosso exemplo, se o usuário final clicar na primeira estrela, a CurrentRowNumber é 0, mas a avaliação do usuário é 1. Então definimos a entrada SelectedStar do evento para CurrentRowNumber + 1 (0+1), que corresponde a avaliação de uma estrela.

* 1. Repita os 2 passos anteriores para a estrela “empty”.

1. Agora que temos o Block dado o gatilho a um evento não obrigatório, o parent deve responder a esse evento ou não. Em User Rating, isso faz sentido que o parenta use a informação sobre a estrela que foi selecionada para salvar a avaliação no banco de dados. Essa operação deve ser possível somente para usuários registrados. Para avaliação média, não precisamos fazer nada por hora.
   1. Abra a tela MovieDetail e selecione a instancia do Block abaixo do texto **Your Rating**.

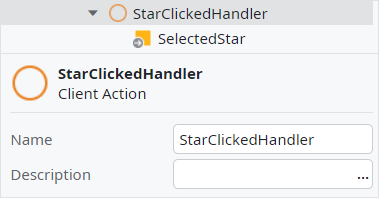


* 1. No **Handler** selecione *(***New Client Action***)*. Nomeie como *StarClickedHandler*.



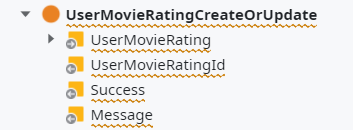
**NOTA:** Essa Action vai executar automaticamente quando o evento for disparado.

* 1. Note que essa Action foi criada com um parâmetro de entrada automaticamente e definida com o valor da **SelectedStar**.

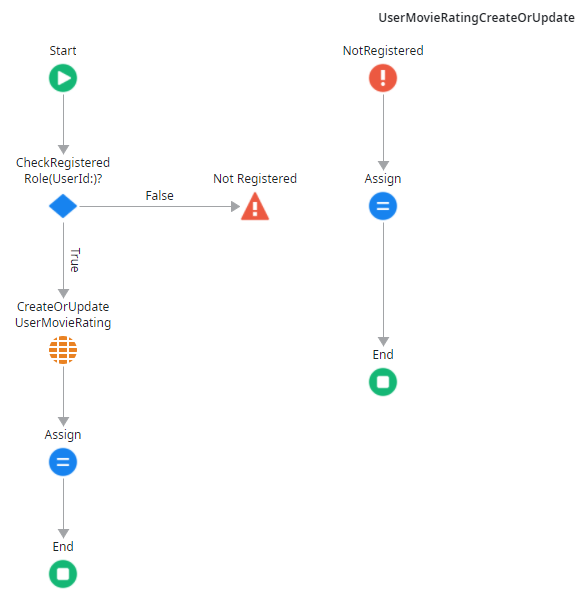


**NOTA:** Os Event Handlers tem acesso aos valores dos parâmetros de entrada dos respectivos Events. Nesse caso, o Handler possui um parâmetro de entrada, no qual vai manter o valor do parâmetro de entrada do Event com o mesmo nome.

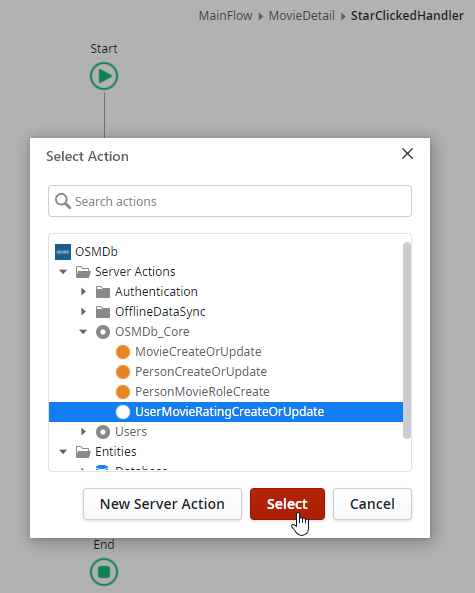
* 1. Quando um usuário avalia um filme, essa avaliação deve ser armazenada no banco de dados. A entidade UserMovieRating vai manter as avaliações de todos os usuários, porém, isso está exposto como read-only. Troque para o módulo **OSMDb\_Core** para criar a lógica para adicionar a avaliação ao banco de dados.
  2. Dentro da aba Logic, crie uma **Server Action** chamada *UserMovieRatingCreateOrUpdate*. Adicione um parâmetro de entrada chamado *UserMovieRating* e três parâmetro de saída, *UserMovieRatingId (UserMovie*Rating Identifier), *Success* (Boolean) e *Message* (Text). Defina a Action como **Public**.



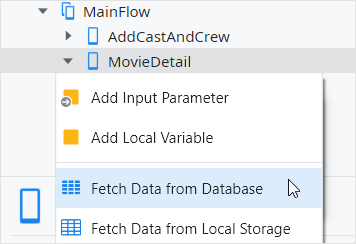
* 1. A lógica da Action deve ser similar as que já tem no módulo OSMDb\_Core. Mas desta vez, a operação deve dar permissão a todos os usuários registrados.



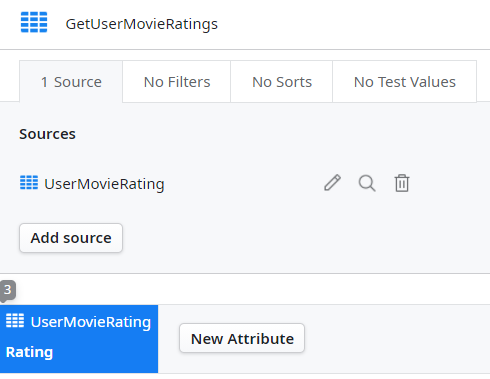
* 1. Publish o módulo. Troque para o módulo OSMDb e referencie essa Action usando o Manage Dependencies.
  2. Dentro da tela MovieDetail, abra a Action **StarClickedHandler**.
  3. Arraste uma **Run Server Action** e solte no fluxo da Action. Selecione a Action **UserMovieRatingCreateOrUpdate**.



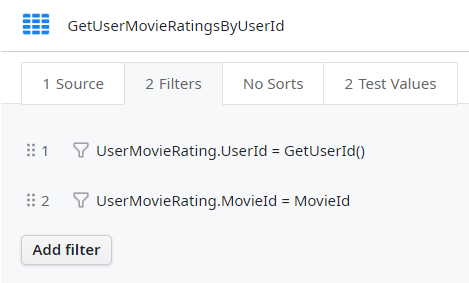
* 1. A Action espera o UserMovieRating como entrada. Como estamos em um cenário onde podemos adicionar uma nova avaliação, ou atualizar uma existente, precisamos buscar a avaliação do banco de dados, se existir. Clique com o botão direito na tela MovieDetail e selecione **Fetch Data from Database**.



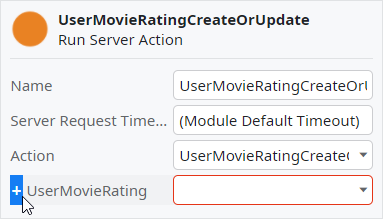
* 1. No Aggregate, defina o **UserMovieRating** como Source do Aggregate.



* 1. Crie dois filtros no Aggregate para pegar a avaliação do usuário logado, usando *GetUserId()*, e para o filme que está sendo mostrado na tela em MovieDetail. Atualize o nome do Aggregate para *GetUserMovieRatingsByUserId* se não for atualizado automaticamente.



* 1. Volte na Action **StarDisplayStarClicked**, selecione o **UserMovieRatingCreateOrUpdate**, expanda o parâmetro de entrada **UserMovieRating** da Action.



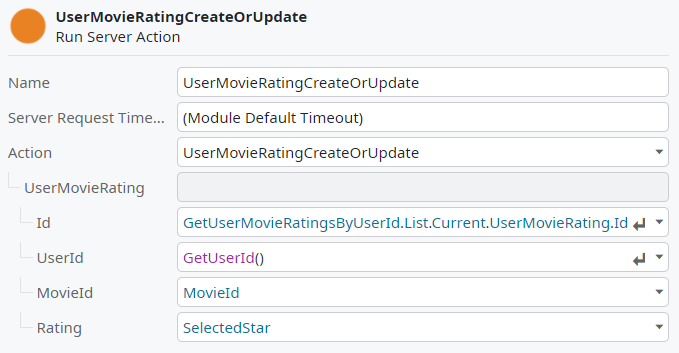
* 1. Defina os atributos do registro UserMovieRating com os seguintes valores

*Id = GetUserMovieRatingsByUserId.List.Current.UserMovieRating.Id*

*UserId = GetUserId()*

*MovieId = MovieId*

*Rating = SelectedStar*

**

**NOTA:** É possível definir individualmente os valores para cada atributo em um registro, como é o que estamos fazendo nesse caso com a entrada UserMovieRating da Server Action.

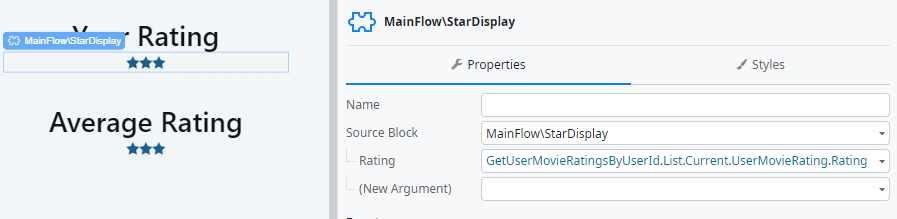
Para o ID, pegamos o registro que foi buscado do banco de dados. Se já existir uma avaliação, vamos usar o mesmo Id, se não tiver, então o Id será 0 e um novo registro será criado no banco.

Para o UserId, definimos para ser o usuário que está logado.

Para o MovieId, usamos o parâmetro de entrada da tela MovieDetail do mesmo nome.

E para o Rating, usamos o valor SelectedStar, que é o parâmetro de entrada do Event do Block StarDisplay.

* 1. Voltando para a tela MovieDetail, selecione novamente a instância do Block abaixo do texto **Your Rating** e mude o valor do Rating para *GetUserMovieRatingsByUserId.List.Current.UserMovieRating.Rating*



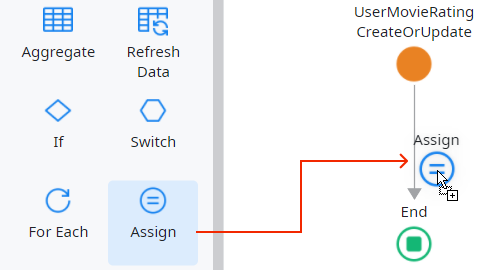
**NOTA:** Agora podemos passar a avaliação do filme atual do bando de dados para o Block. A Avaliação é buscada no Aggregate criado acima, ao invés do valor estático.

1. Vamos testar isso no navegador ao avaliarmos um filme. Vamos notar que a avaliação não muda, significando que o valor correto de estrelas preenchidas não aparece. Vamos consertar isso garantindo que quando uma avaliação é dada, as estrelas atualizem automaticamente na tela.
   1. Publish o módulo e abra no navegador.
   2. Faça o login com algum dos seus usuários.
   3. Selecione um filme que não tenha uma avaliação e clique em um número de estrelas. Elas aparecem preenchidas? A resposta deve ser não...



**NOTA:** Isso ocorre porque salvamos a avaliação no banco de dados, porém duas coisas estão faltando: a avaliação buscada do banco de dados (com o Aggregate) ainda não reflete essas mudanças e o Block “não sabe” que essa avaliação mudou. No caso do Block, não se esqueça que as estrelas dependem da List, que ainda não mudou, por isso as estrelas ainda não mudaram.

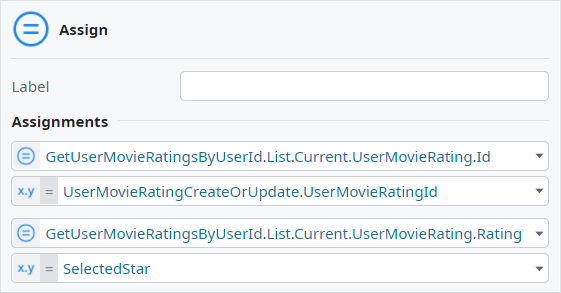
1. Vamos resolver este problema acima, atualizando a informação sobre a avaliação e atualizando a List de estrelas dentro do Block.
   1. Abra a Action StarClickedHandler, arraste um Assign e solte depois da Action **UserMovieRatingCreateOrUpdate**



* 1. Defina as seguintes atribuições

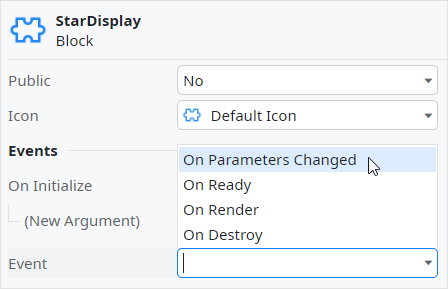
*GetUserMovieRatingsByUserId.List.Current.UserMovieRating.Id = UserMovieRatingCreateOrUpdate.UserMovieRatingId*

*GetUserMovieRatingsByUserId.List.Current.UserMovieRating.Rating = SelectedStar*

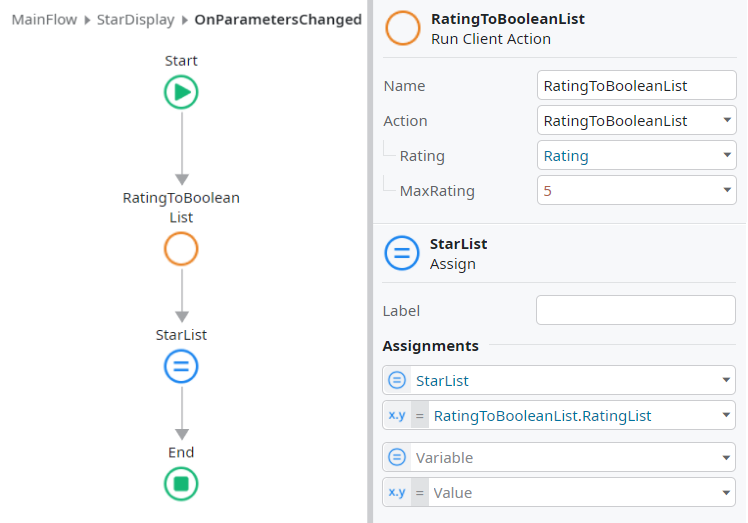
**

**NOTA:** Isso atualiza o registro do UserMovieRating buscado do Aggregate com o novo Id (se a avaliação não existir) e com a nova avaliação.

* 1. Abra o Block StarDisplay e nas propriedades do evento, selecione **On Parameters Changed**



* 1. Isso cria uma nova Client Action dentro do Block, *OnParametersChanged*. Essa Action deve recalcular a variável **StarList**, usando a Action **RatingToBooleanList**. A lógica é similar a da Action **OnInitialize**.



**NOTA:** O On Parameters Changed Event é automaticamente disparado quando um parâmetro de entrada do Block é modificado no parent. Como mudamos a avaliação no passo anterior, o parâmetro de entrada do Rating do Block terá seu valor mudado e o evento será automaticamente disparado. Na lógica, executamos o RatingToBooleanList com a nova avaliação, resultando numa nova StarList, no qual irá mostrar a quantidade correta de estrelas coloridas no Block.

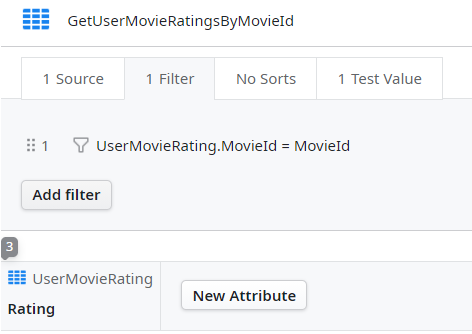
* 1. Publique o módulo e teste a aplicação no navegador. A avaliação aparece corretamente agora?



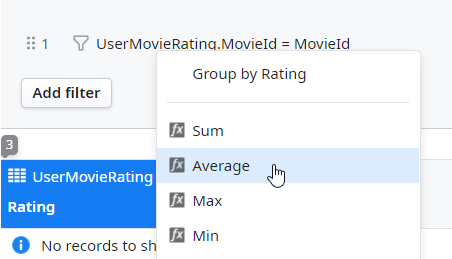
## Avaliação Média

Agora que temos a avaliação de filmes do usuário implementada, precisamos lidar com a avaliação média. Primeiramente, precisamos calcular a avaliação média para todos os filmes e passar essa informação como parâmetro de entrada do Block. Isso vai indicar quantas estrelas coloridas devem aparecer nessa instância particular do Block na tela MovieDetail. Depois, precisamos ter certeza de que essa avaliação média de estrelas não estão clicáveis (somente as avaliações de usuários devem ser clicáveis).

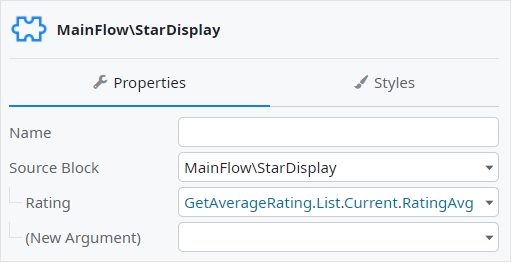
1. Busque do banco de dados a avaliação média de votos de um filme em particular e passe essa informação para o Block StarDisplay.
   1. Clique com o botão direito na tela MovieDetail e selecione **Fetch Data from Database**.
   2. No novo Aggregate, selecione a entidade **UserMovieRating** como source.
   3. Adicione um novo Filter para garantir que estamos buscando avaliações somente deste filme



* 1. Selecione a coluna Rating no Aggregate e escolha a opção **Average**.



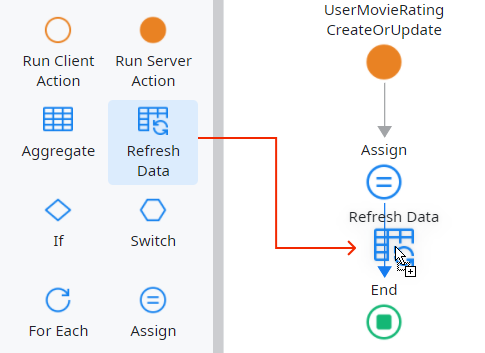
* 1. Mude o nome do Aggregate para *GetAverageRating*
  2. Abra a tela MovieDetail novamente e selecione o Block abaixo do texto **Average Rating** e mude o valor do parâmetro de entrada rating para *GetAverageRating.List.Current.RatingAvg*



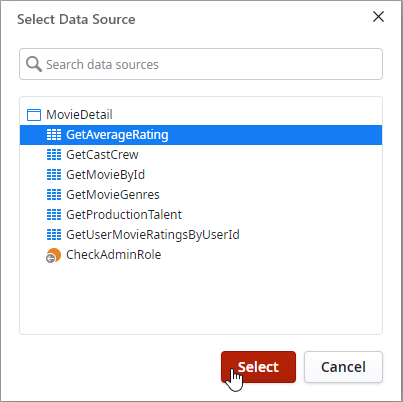
* 1. Publish o módulo e teste a aplicação no navegador.
  2. Faça login com uma conta de usuário diferente da anterior e avalie o mesmo filme. A avaliação média mudou? Não né.

**NOTA:** Pela mesma razão da avaliação do usuário, o Aggregate buscou a avaliação média no tempo. Com a nova avaliação, precisamos nos certificar de que a média é recalculada, assim o Block mostrará o número correto de estrelas.

* 1. Abra a Action **StarClickecHandler**. Arraste o nó **Refresh Data** e solte abaixo do Assign.



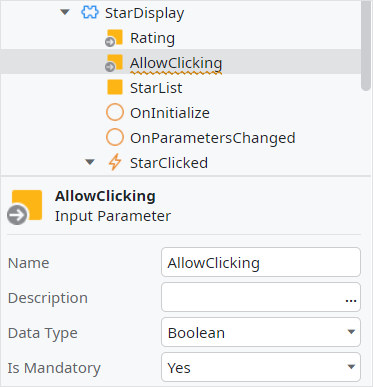
* 1. Na nova janela, selecione o aggregate **GetAverageRating**



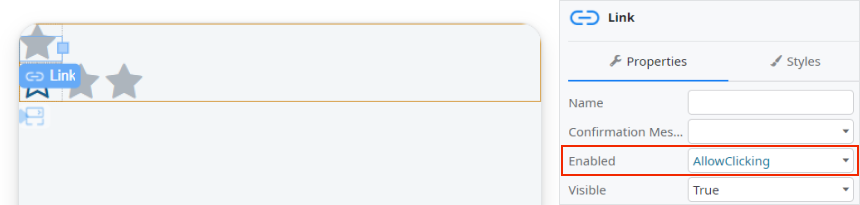
**NOTA:** O nó Refresh Data executa um Aggregate on demand. Nesse caminho, quando um evento é disparado, a avaliação é salvada no banco de dados e a média é recalculada novamente.

* 1. Publique o módulo e repita o mesmo teste acima. A avaliação média deverá estar funcionando bem.

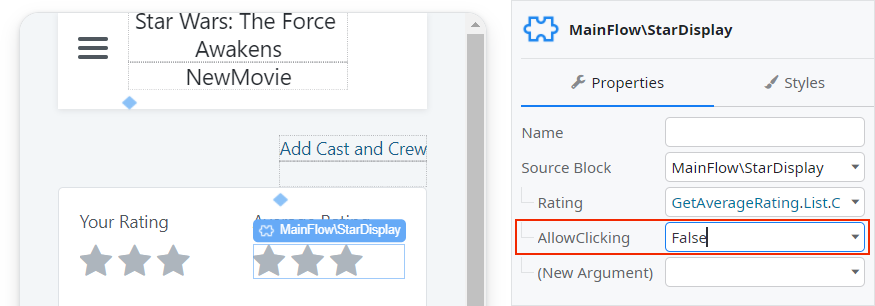
1. Para assegurar que somente o Block da avaliação do usuário é clicável, precisamos de um parâmetro de entrada adicional no Block, *AllowClicking*. Isso nos ajudará a definir quando os Links nas estrelas serão clicáveis ou não. Assim, para a avaliação de usuários, somente usuários registrados podem clicar nas estrelas, enquanto na de média, ninguém poderá clicar nas estrelas.
   1. Clique com o botão direito no Block StarDisplay e selecione o **Add Input Parameter**. Nomeie como *AllowClicking* e se assegure que o **Data Type** é do tipo *booleano*.



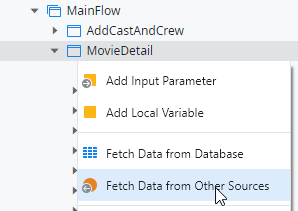
* 1. Selecione o **Link** que circunda a estrela preenchida e defina sua propriedade **Enabled** para *AllowClicking*.



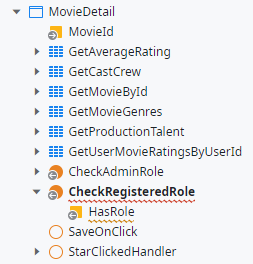
* 1. Repita o mesmo processo para o outro Link (empty star).
  2. Abra a tela MovieDetail e selecione o Block **Average Rating**. Defina sua propriedade **AllowClicking** para *False*.



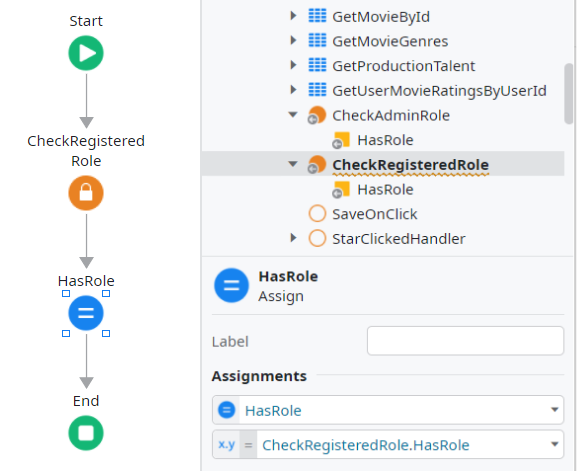
* 1. Para a avaliação do usuário, precisamos de um pouco a mais, já que precisamos verificar se o usuário logado no momento é um usuário registrado. Clique com o botão direito na tela MovieDetail e selecione **Fetch Data from Other Sources**

****

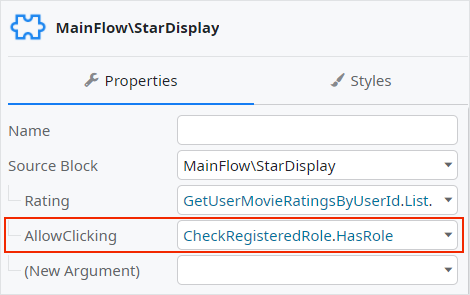
* 1. Defina o nome da Data Action para *CheckRegisteredRole* e mude seu parâmetro de saída para *HasRole*, com **Data Type** definido para *Boolean*.



* 1. Defina a lógica da Data Action para verificar se o usuário logado possui role registrada e atribua a saída do HasRole com o resultado da Action CheckRole.



* 1. Volte a tela MovieDetail, selecione o Block User Rating e defina o parâmetro de entrada do **Allow Clicking** para *CheckRegisteredRole.HasRole*

**

* 1. Publique o módulo e teste a aplicação no navegador.