Criando um Reactive Web App

Exercício – Como fazer

# 

Resumo1

[**Como**](#_3qrc8ufe22au) **fazer** 1

Criar Aplicação 1

[Criar um Client Action](#_kcq8u66lvwmg) 5

Usar a Client Action na tela 8

# 

# Resumo

Neste exercício nós iremos criar nossa primeira aplicação web reativa. Esta aplicação irá gerenciar filmes e pessoas envolvidas nesses filmes, como elenco e equipe.

Através dos próximos exercícios, iremos progressivamente desenvolvendo a aplicação com novas funcionalidades. Por hora, iremos começar criando a aplicação e dois módulos, um para a interface e outro para a lógica e entidades.

* **Reactive Web App Name:** OSMDb\_<your\_initials>
* **Interface module:** OSMDb\_<your\_initials> - Reactive Web App module
* **Core module:** OSMDb\_Core\_<your\_initials> - Blank module

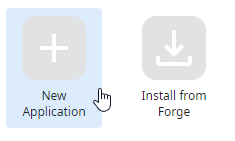
Depois que a aplicação é criada, precisamos criar uma ação que atribui um texto estático a uma variável no módulo em branco. Após isso, vamos mostrar em tela essa mensagem no módulo Reactive Web.

# Como fazer

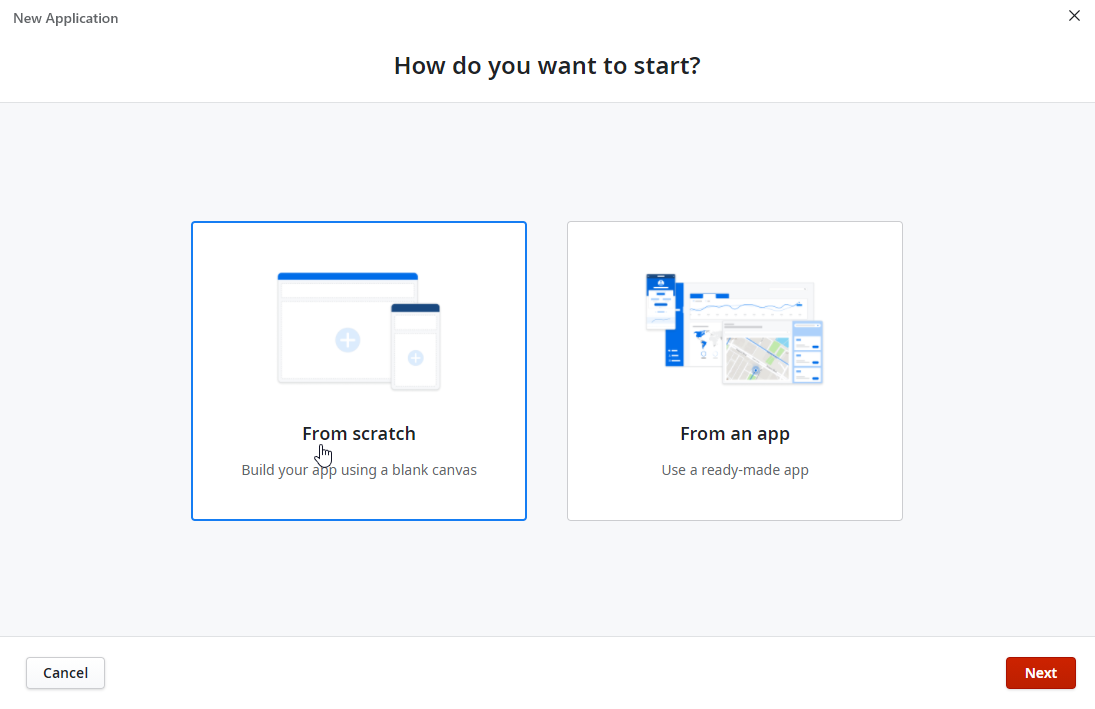
Neste documento, vamos descrever o passo a passo o exercício 1 – Criando uma aplicação Reactive Web.

## Criar Aplicação

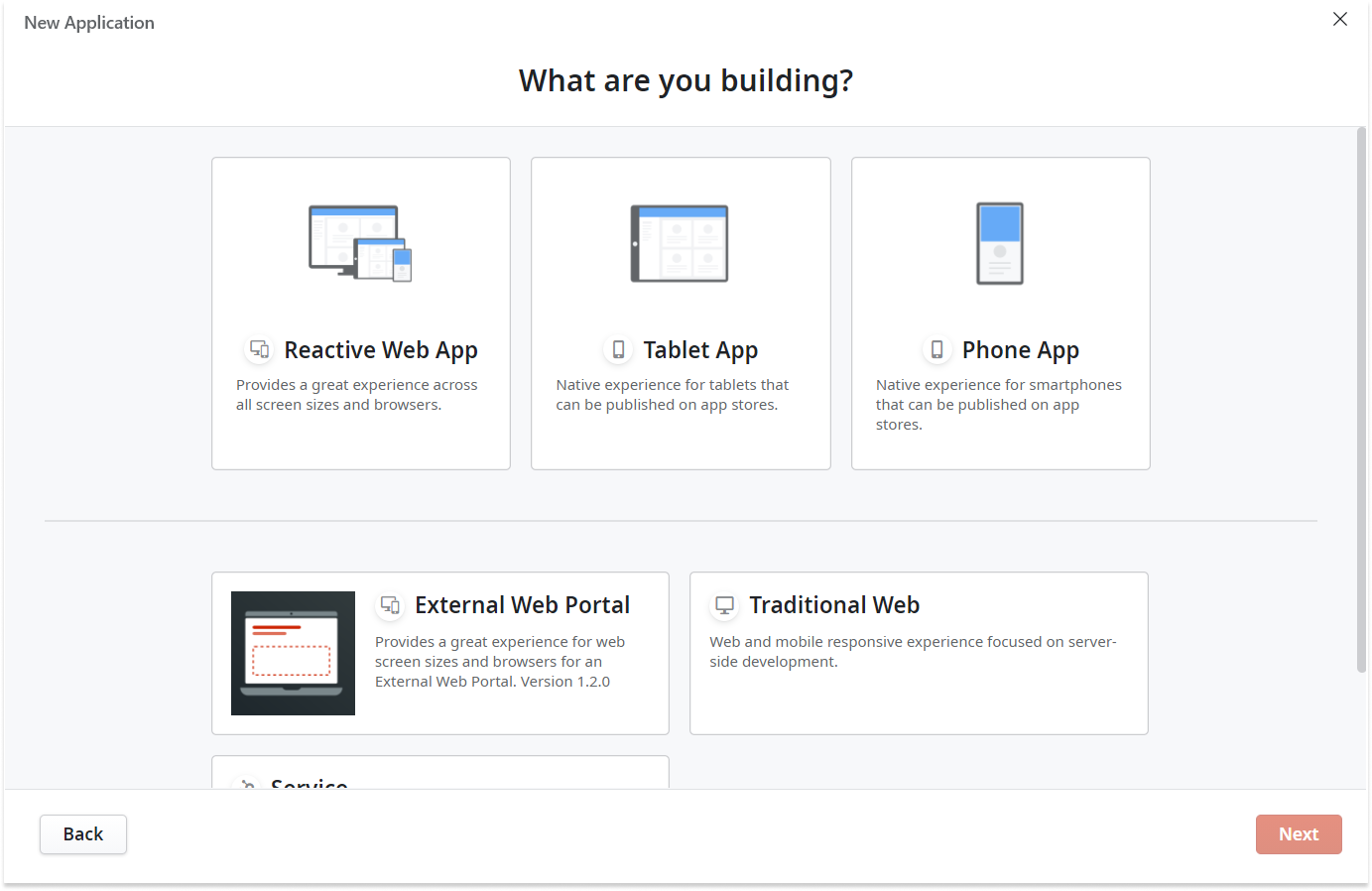
1. Crie a aplicação Reactive Web chamada *OSMDb*.
   1. No Service Studio, clique no ícone **New Application**.



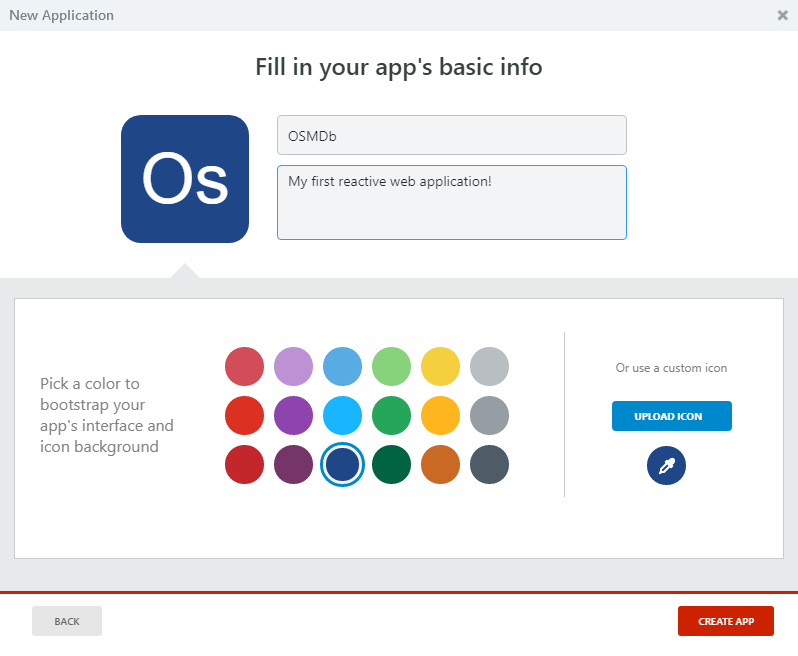
* 1. Escolha a opção **From scratch**, já que queremos desenvolver a aplicação do início ao fim.



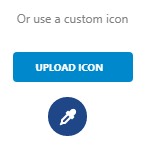
* 1. Escolha a opção **Reactive Web App**.



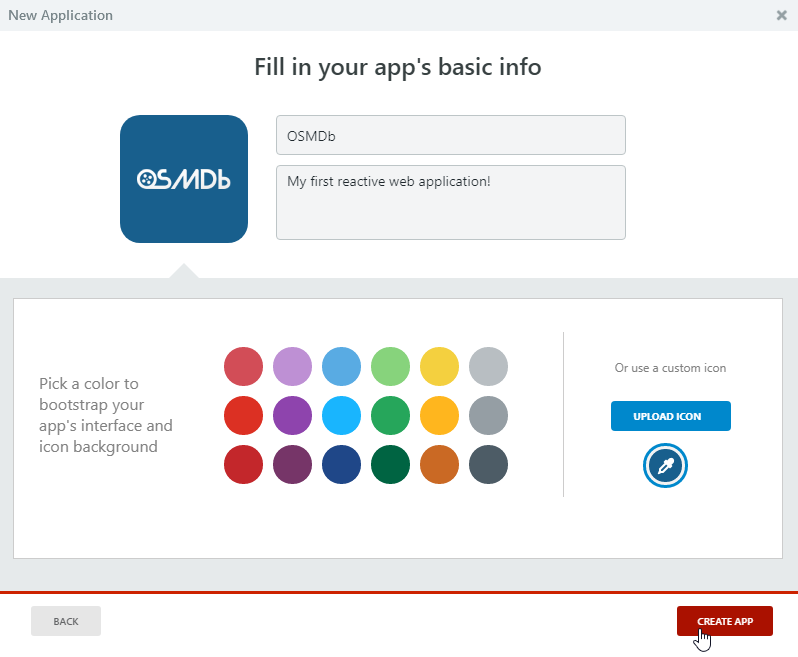
* 1. Preencha o nome e descrição da Aplicação. Como estamos usando um servidor compartilhado, insira suas iniciais no nome da aplicação: *OSMDb\_<suas\_iniciais>*.



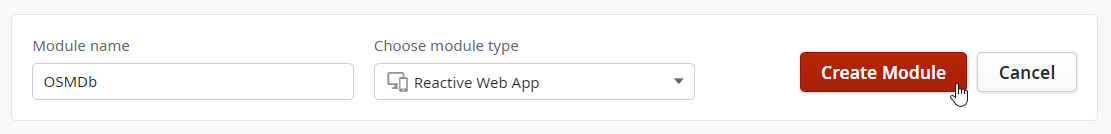
* 1. Clique no **Upload Icon** e selecione o arquivo *OSMDb-icon.png* para usar o ícone da aplicação customizado.



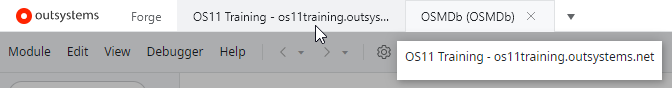
* 1. Clique em **Create App** para criar a aplicação.



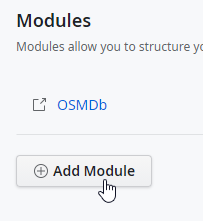
1. Crie dois módulos para a aplicação: o módulo **Reactive Web App** chamado *OSMDb\_<Suas\_Iniciais>* e um módulo em branco (blank) chamado *OSMDb\_Core\_<Suas\_Iniciais>*.
   1. Na tela da aplicação, já existe uma sugestão de um módulo. Confira que se chama *OSMDb*, do tipo **Reactive Web App** e selecione **Create Module**.



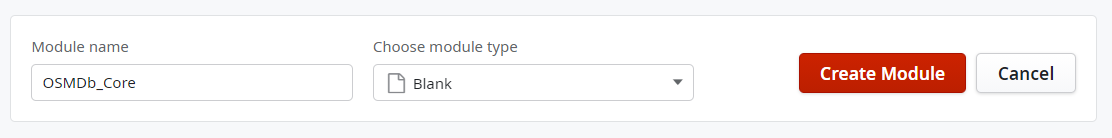
* 1. Isto abrirá uma nova aba com o módulo *OSMDb* no Service Studio. Volte para a seção anterior selecionando a aba com a informação do ambiente.



* 1. Clique em **Add Module** e defina o novo módulo para ser *OSMDb\_Core\_<Suas\_Iniciais>* e do tipo **Blank**.



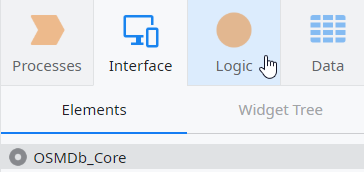
* 1. Clique em **Create Module**.



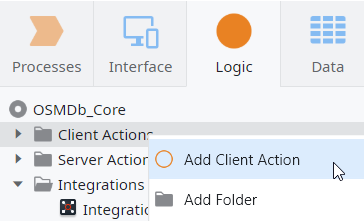
**NOTA:** Deste ponto em diante, a aplicação e os nomes dos módulos serão referenciados sem as iniciais por simplificação.

## Criar uma Client Action

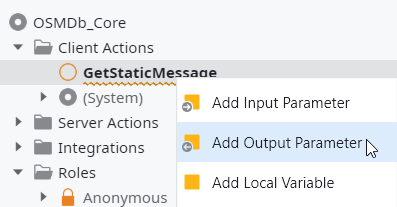
1. No **OSMDb\_Core**, crie uma Client Action chamada *GetStaticMessage*, com o parâmetro de saída chamado *Message*.
   1. No módulo **OSMDb\_Core**, mude para a aba Logic do Service Studio.



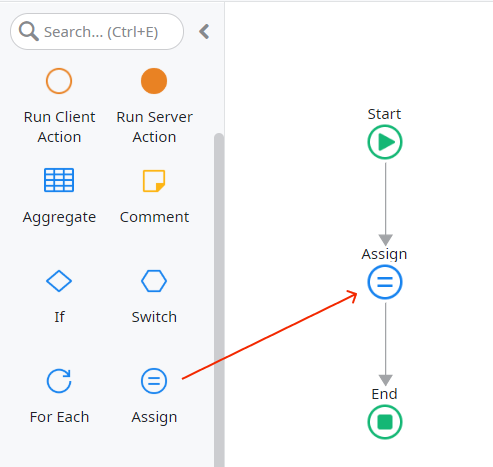
* 1. Clique com o botão direito na pasta Client Actions e selecione **Add Client Action**.



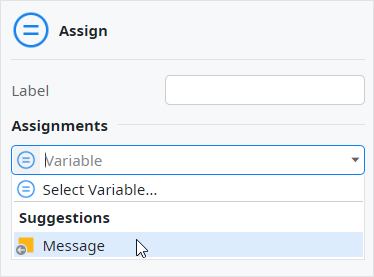
* 1. Configure seu nome como *GetStaticMessage*.
  2. Clique com o botão direito na Action e selecione a opção **Add Output Parameter**.Configure seu nome como *Message.*



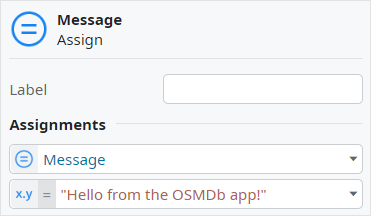
1. Defina a lógica da Client Action. A Action deverá atribuir uma mensagem de texto estático no parâmetro de saída **Message**. Também, configure a Action como **Public** para ser usada em outros módulos e como uma **Function** para ser usado diretamente nas telas.
   1. Arraste um **Assign** e solte no fluxo da Action.



* 1. Clique no Assign para selecioná-lo. Na área de propriedades, abaixo de Assignments, selecione o parâmetro de saída **Message**.

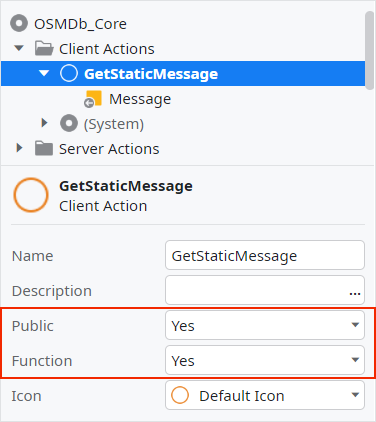


* 1. Abaixo do **Value**, digite uma mensagem estática como *“Hello from the OSMDb app”*.

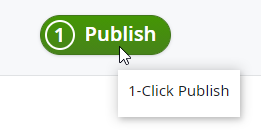


**NOTA:** Assim é como criamos uma simples atribuição em OutSystems e neste caso, nós temos: Message = “*Hello from the OSMDb app*”

* 1. Clique na Action **GetStaticMessage** e em suas propriedades, e configure a opção **Public** para *Yes* e a opção **Function** para *Yes*.



* 1. Publique o módulo OSMDb\_Core para salvar a mudanças no servidor.



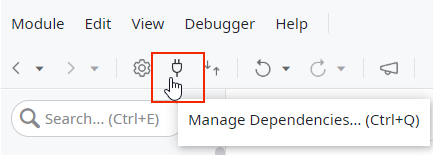
## Usar um Client Action na tela

A Client Action **GetStaticMessage** foi criada no módulo OSMDb\_Core, porém, queremos usá-la no módulo OSMDb para mostrar o texto estático na tela.

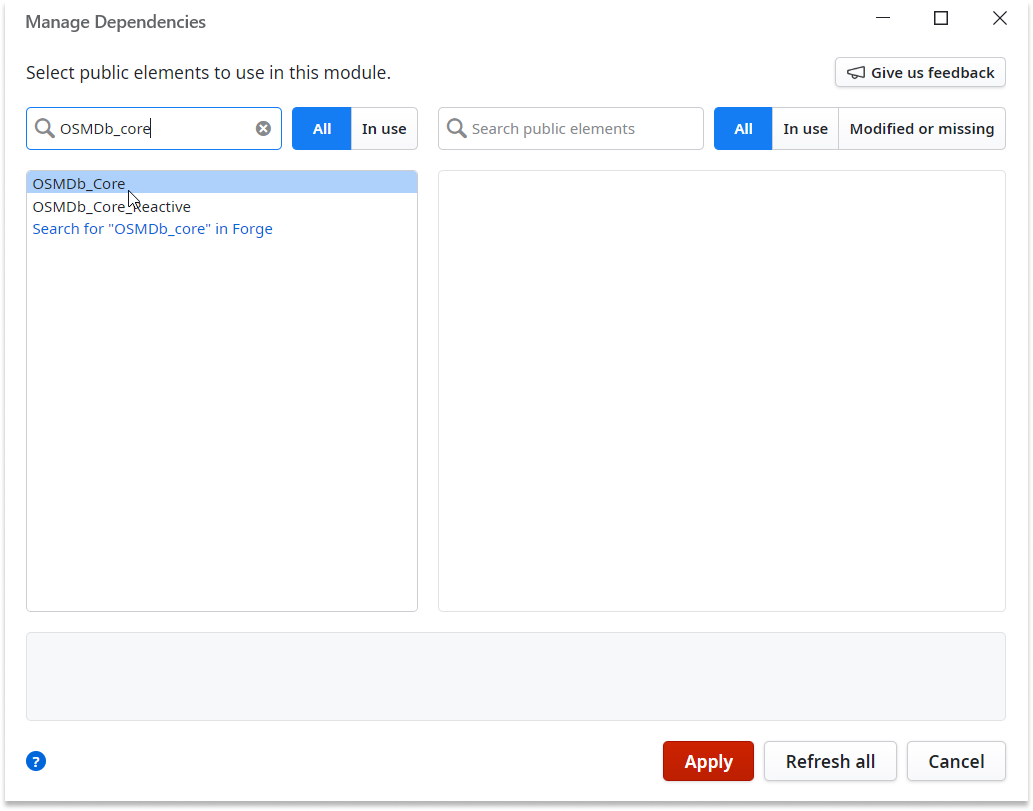
1. Referencie a Client Action **GetStaticMessage** do módulo OSMDb.
   1. Alterne para o módulo **OSMDb**.



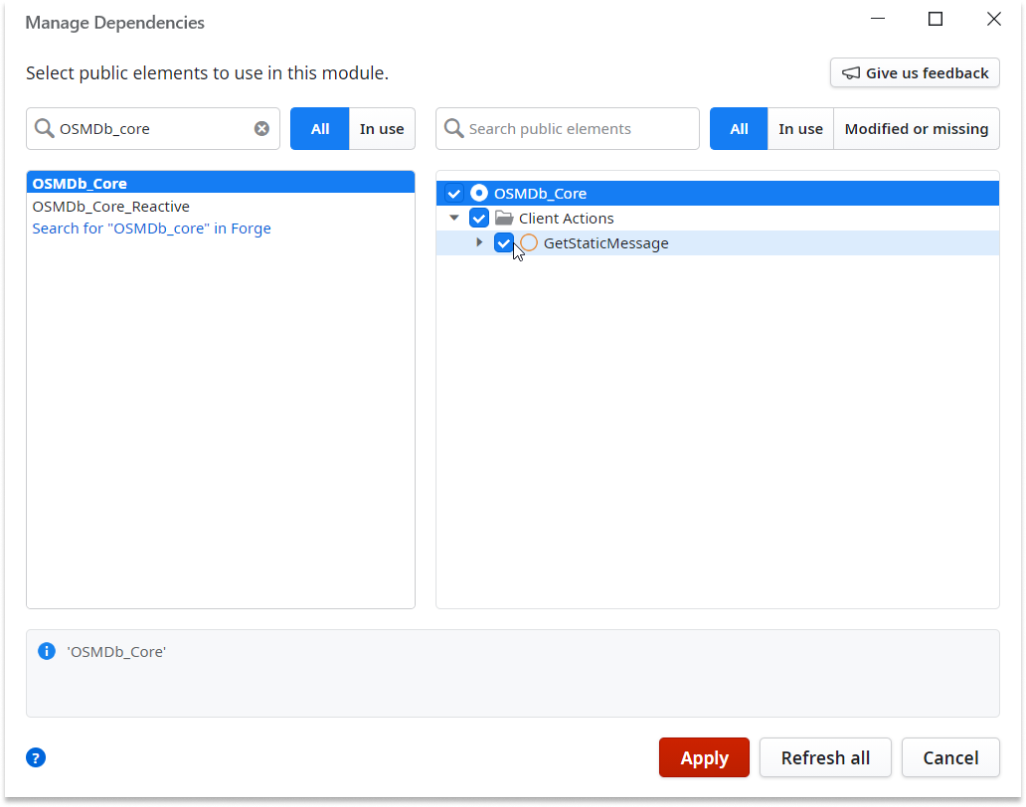
* 1. Abra o **Manage Dependencies**.



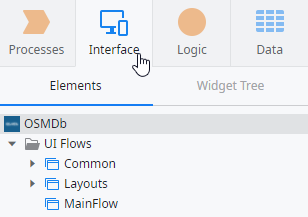
* 1. Uma nova janela irá aparecer com todos os módulos que possuem elementos públicos no ambiente. Procure pelo módulo **OSMDb\_Core** e clique nele para selecioná-lo.



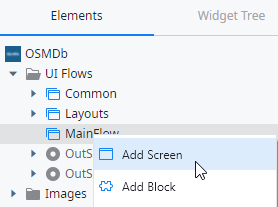
* 1. Selecione a Action **GetStaticMessage** na direita. Clique em **Apply** para sair. Isto irá referenciar a Action criada no módulo Core.



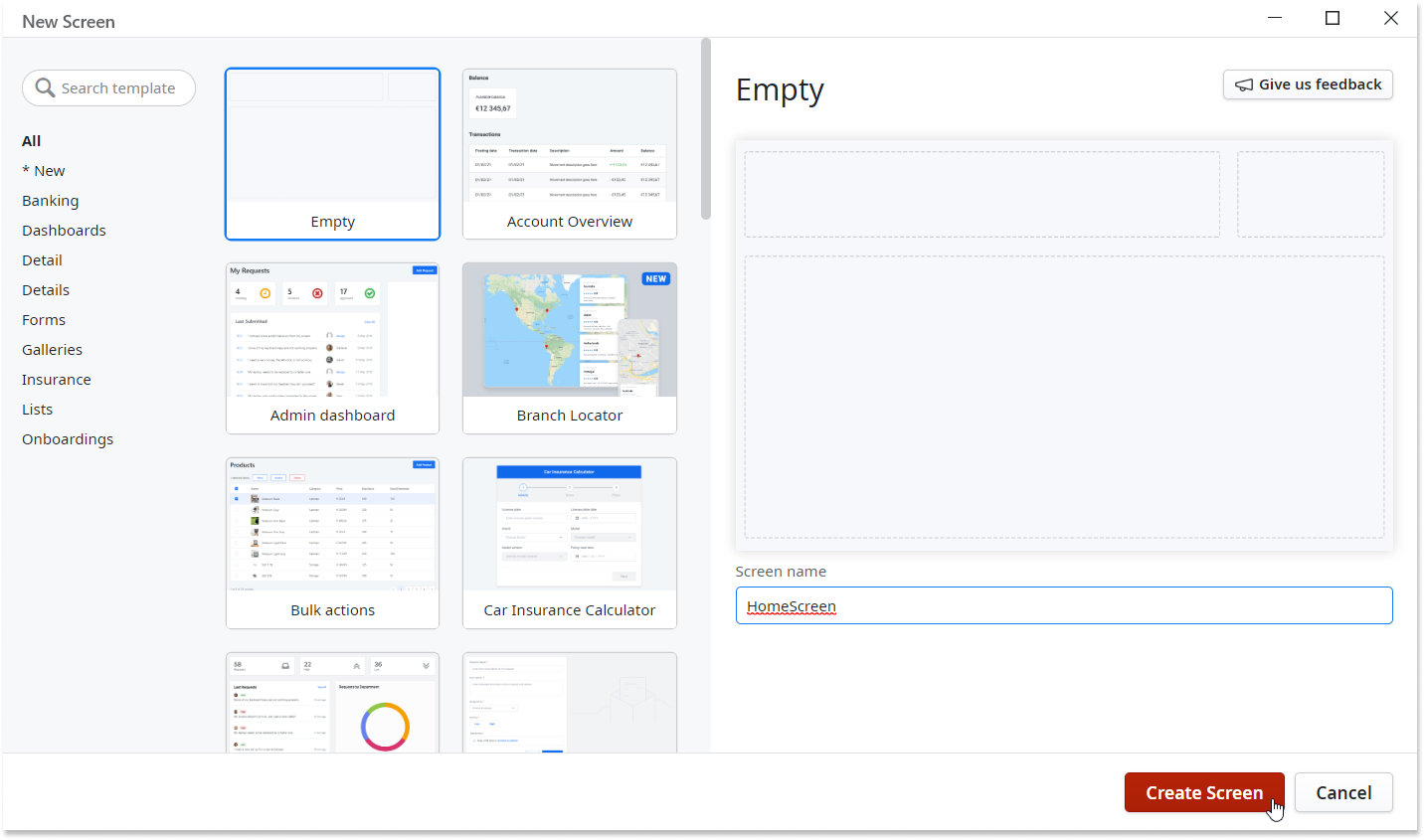
1. Crie uma nova tela chamada *HomeScreen*. Esta tela deve mostrar apenas a mensagem estática na saída da Client Action definida acima.
   1. ALterne para aba **Interface** no Service Studio.



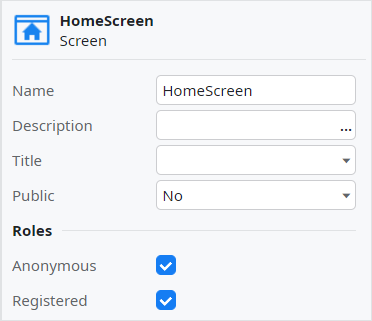
* 1. Clique com o botão direito em MainFlow e selecione a opção **Add Screen**.



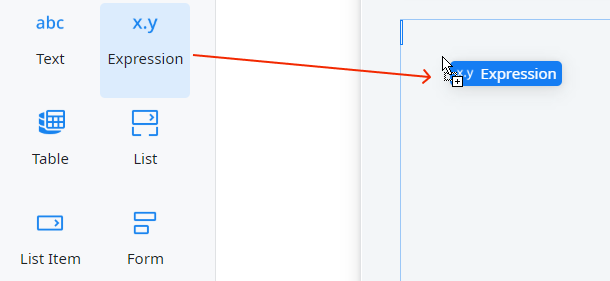
* 1. Selecione o modelo vazio **Empty** template já que queremos uma tela vazia. Defina seu nome como *HomeScreen* e clique em **Create Screen** para finalizar.



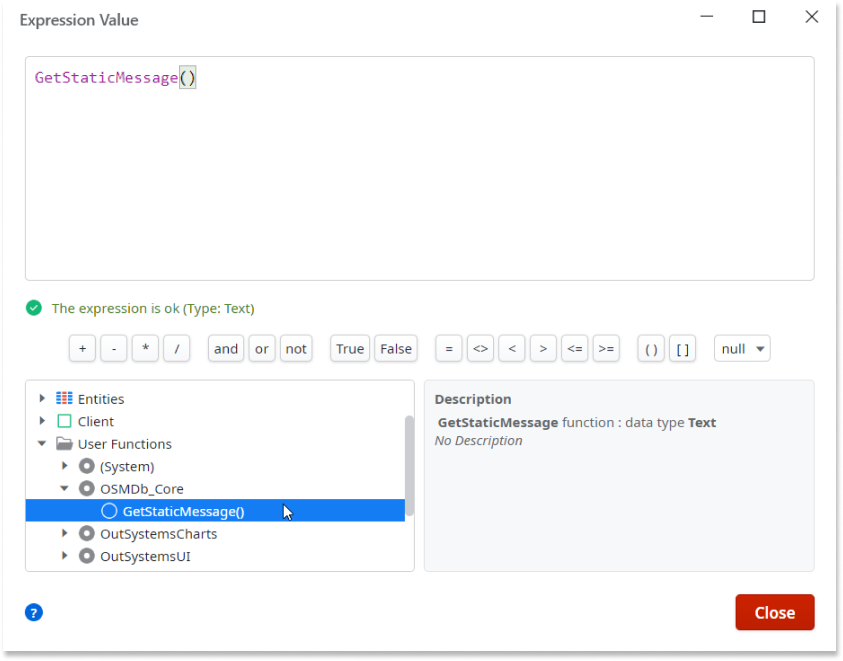
* 1. Configure a tela como **Anonymous**, assim os usuários poderão acessá-la sem a necessidade de fazer o login.

****

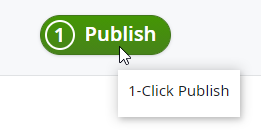
* 1. Arraste uma **Expression** para a tela.



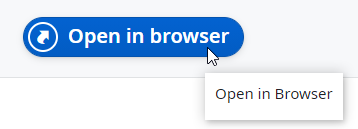
* 1. Na janela que aparece, abra a pasta User Functions e a OSMDb\_Core e selecione a Client Action **GetStaticMessage()**. Clique em **Done** para sair.



* 1. Publique o módulo clicando em 1-click publish button.



* 1. Abra a aplicação no navegador.



* 1. Confira que a mensagem aparece na tela.