Security and Exceptions

Exercício – Como Fazer

**Esboço**

**Como fazer** 1

Criar Usuários e Cargos (Roles) 1

Verificar Cargos (Roles) na Lógica 3

Acesso Restrito nas Telas 9

Entradas Desabilitadas nas Telas

# 

# Esboço

Nesse exercício, adicionaremos alguns componentes de segurança em nossa aplicação. Até agora, todas as telas podem ser acessadas e as operações podem ser completadas com um cargo (role) anônimo, significando que não existem restrições de segurança. Ao final desse exercício, queremos nos assegurar que somente administradores poderão criar/atualizar informações para os filmes e pessoas, incluindo a edição de elenco ou membro da equipe para um filme.

Queremos fazer isso em diversos passos:

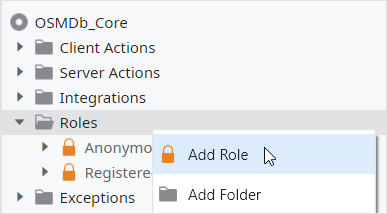
1. Criar OSMDbAdmin role.
2. Criar alguns usuários finais e conceder o cargo (role) para um deles,
3. Garantir na lógica que somente um administrador pode adicionar um membro do elenco ou equipe em um filme. Use as Exceptions para mostrar uma mensagem de erro para o usuário caso ele não tenha as permissões corretas
4. Filtrar o acesso para a tela AddCastAndCrew para os usuários OSMDbAdmin.
5. Ter certeza de que somente usuários administradores podem editar as informações dos filmes e pessoas.

# Como Fazer

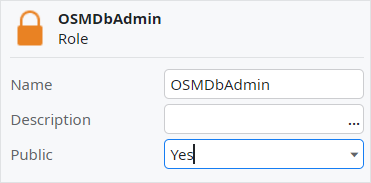
Nessa seção, descreveremos o passo a passo do exercício *6 – Segurança*.

## Criar Usuários e Cargos (Roles)

1. Crie uma nova *OSMDbAdmin* role, no módulo OSMDb\_Core.
   1. Abra a aba Logic no módulo **OSMDb\_Core**.
   2. Com o botão direito, clique na pasta roles e escolha **Add Role**



* 1. Nomeie para *OSMDbAdmin* e o torne **Public**



* 1. **Publish** o módulo.

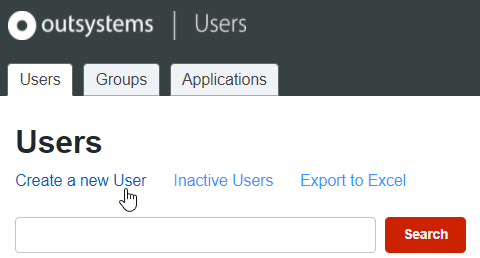
1. Vá para a aplicação Users e crie dois novos usuários: um com a OSMDbAdmin role e o outro sem.
   1. Abra a aplicação Users no navegador, navegando para *https://<environment>/Users*
   2. Se você está usando um ambiente compartilhado em um BootCamp, peça login com as seguintes credenciais:

**Username:** *admin*

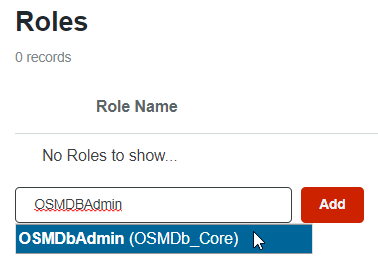
**Password:** *outsystems*

Se você está usando um ambiente pessoal, apenas faça o login com suas credenciais no OutSystems.

* 1. Clique em **Create a new User** (Criar novo usuário).



* 1. Preencha os dados obrigatórios com qualquer informação que você queira e crie uma senha simples, por exemplo *123456* e clique em salvar.
  2. Mova para abaixo para a área Roles. No dropdown, procure por **OSMDBAdmin** role (Lembre-se de que ele deve estar associado com seu módulo central) e clique em Add.

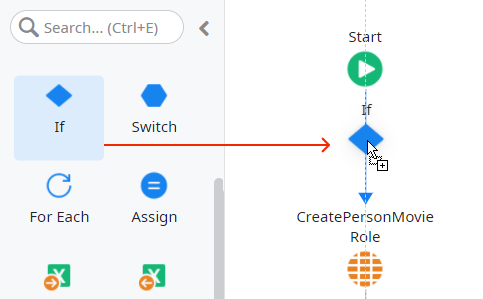


* 1. Crie um usuário, mas dessa vez sem qualquer role associado a ele.

## Verificar Cargos (roles) na Lógica

Agora que a aplicação possui um novo role, vamos usá-lo para expandir a lógica e nos certificar que somente usuários com o OSMDbAdmin role possam criar/atualizar registros no banco de dados.

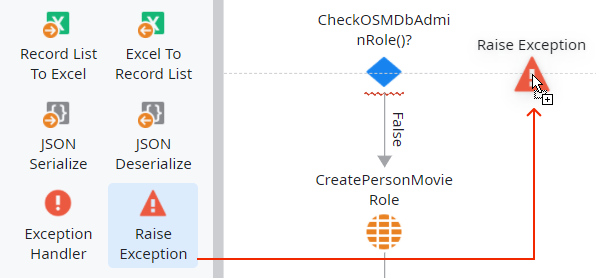
1. No Módulo OSMDb\_Core, mude a Action **PersonMovieRoleCreate** para nos certificar que somente usuários com o OSMDbAdmin role possam criar um registro no banco de dados. A Action deve retornar se a operação foi bem-sucedida ou não com uma mensagem de sucesso/erro.
   1. Abra a Action PersonMovieRoleCreate, arraste um If e solte depois do nó Start.



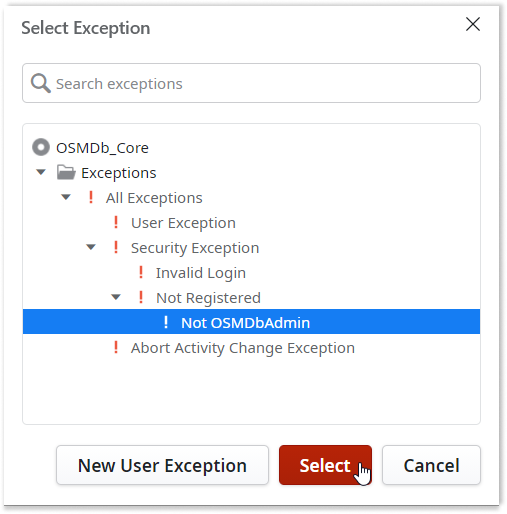
* 1. Defina a **Condition** do If para *CheckOSMDbAdminRole()*

**NOTA:** A Action CheckRole pode ser usada nas Server Actions para verificar se um determinado usuário tem um role específico. Quando a Action CheckRole é chamada sem nenhum parâmetro de entrada, a verificação é feita considerando o usuário que está logado.

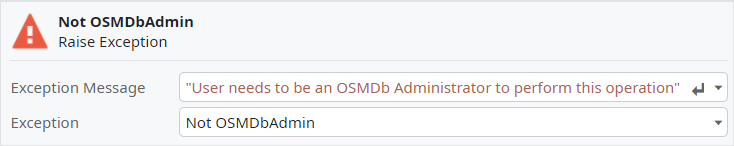
* 1. Arraste a **Raise Exception** e solte perto do If.



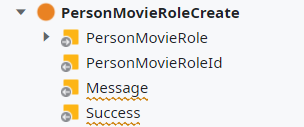
* 1. Na nova janela, escolha a Exception **OSMDbAdmin**.



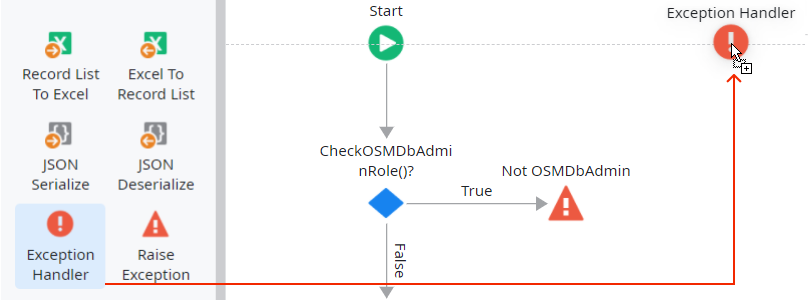
* 1. Defina a propriedade **Exception Message** para *"User needs to be an OSMDb Administrator to perform this operation" (O Usuário precisa ser um Administrador OSMDb para realizar essa operação).*

**

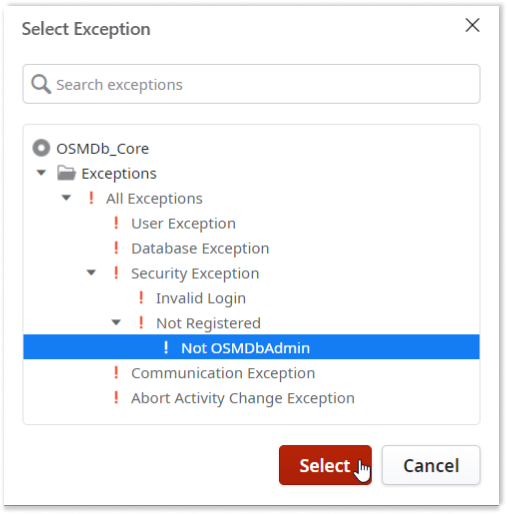
* 1. Conecte o If ao nó do Raise Exception.
  2. Adicione um novo parâmetro de saída na Action PersonMovieRoleCreate e nomeie como *Success*. Se assegure que o **Data Type** seja booleano.
  3. Adicione outro parâmetro de saída na Action e nomeie como *Message.*

**

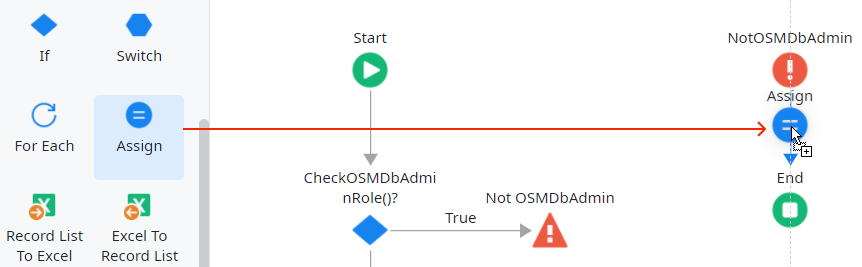
* 1. Arraste um **Exception Handler** (manipulador de exceção) e solte próximo ao fluxo principal da Action.



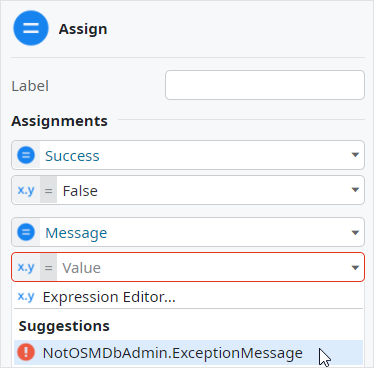
* 1. Nessa janela de Exception, selecione a opção **Not OSMDbAdmin**.



* 1. Arraste um **Assign** e solte no fluxo da Exception



* 1. Defina as seguintes atribuições:



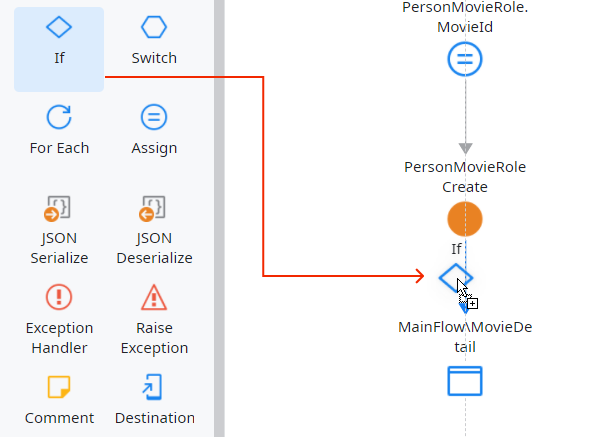
**NOTA:** Uma mensagem definida no nó Raise Exception pode ser usada diretamente para referenciar a Exception e o nome da propriedade (ExceptionMessage).

* 1. Clique com o botão direito no If e selecione **Swap Connectors.**
  2. Selecione um Assign bem antes do nó End no fluxo principal da Action e selecione as seguintes assignments:

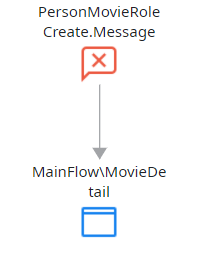
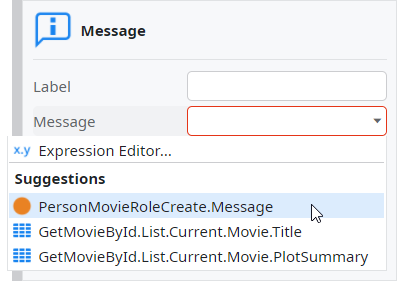
*Success = True*

*Message = "Record created successfully"*

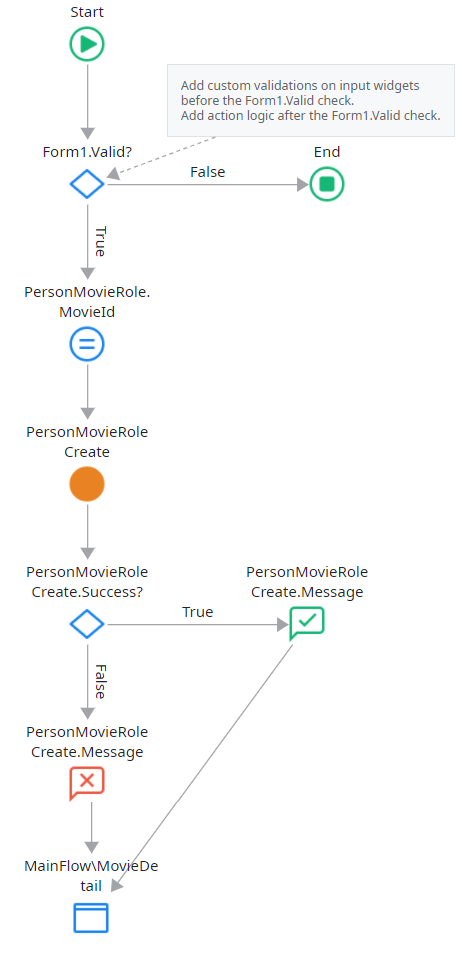
1. Repita exatamente esses passos para as Actions **MovieCreateOrUpdate** e **PersonCreateOrUpdate**.
2. Publish o módulo.
3. Troque para o módulo OSMBb e atualize as dependências (refresh the dependencies).
4. Na Action SaveOnClick da tela AddCastAndCrew, use os parâmetros de saída da Action PersonMovieRoleCreate para criar a lógica para mostrar a mensagem de retorno ao usuário final.
   1. Abra a Action **SaveOnClick** na telaAddCastAndCrew.
   2. Arraste um **If** e solte abaixo da Action PersonMovieRoleCreate.



* 1. Defina a **Condition** do If como *PersonMovieRoleCreate.Success*
  2. Arraste a **Message** e solte abaixo do If. Defina a propriedade da **Message** para PersonMovieRoleCreate na saída da Message e o **Type** para Error.



* 1. Arraste uma Mensagem e solte à direita do If. Conecte o If para essa nova mensagem, depois essa nova mensagem na Destination da tela MovieDetail.
  2. Defina a propriedade **Message** para ser o parâmetro de saída da Action PersonMovieRoleCreate e o **Type** *Success*. O fluxo deve ter a aparência da imagem abaixo:



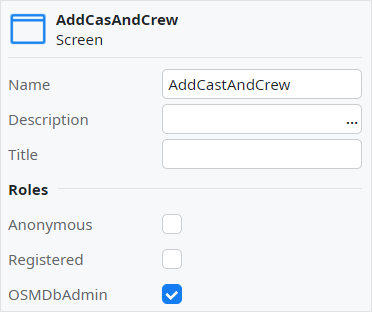
1. Repita a mesma lógica acima para as Actions **SaveOnClick** nas telas **MovieDetail** e **PersonDetail**.

## Acesso Restrito nas Telas

Na seção anterior, adicionamos uma lógica para verificar o role de um usuário em um fluxo. Nessa seção, vamos usar os roles para restringir o acesso às telas ou à alguma funcionalidade de uma tela.

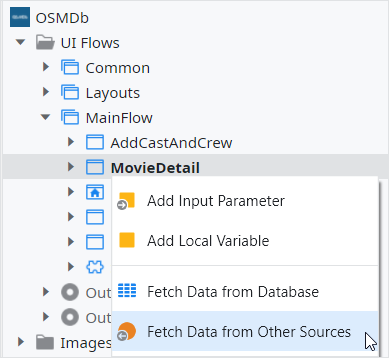
Vamos começar com as Telas: Somente usuários OSMDbAdmin podem acessar a tela AddCastAndCrew. Depois, o link *Add Cast and Crew to Movie* emMovieDetail deve estar oculto para todos os usuários que não tenham OSMDbAdmin role.

1. Selecione a tela AddCastAndCrew e na área de propriedades, dentro de Roles, remova o acesso das roles **Anonymous** e **Registered**.



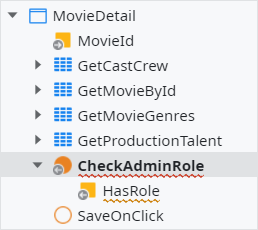
**NOTE:** Certifique-se de que a função OSMDbAdmin seja referenciada a partir do módulo Core, caso contrário, não aparecerá nas propriedades 'Tela'.

1. Agora, precisamos também mostrar o link **Add Cast and Crew** na tela MovieDetail somente para usuários OSMDBAdmin.
   1. Clique com o botão direito na tela MovieDetail e selecione **Fetch Data from Other Sources** (buscar dados de outras fontes).

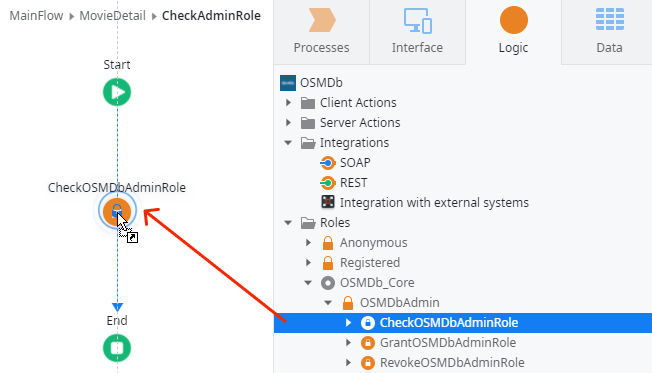


**NOTA:** Como a Action CheckRole é uma Server Action, precisamos usar uma Data Action para determinar se o usuário possui a role esperada.

* 1. Nomeie como *CheckAdminRole* e mude o parâmetro de saída para *HasRole.*

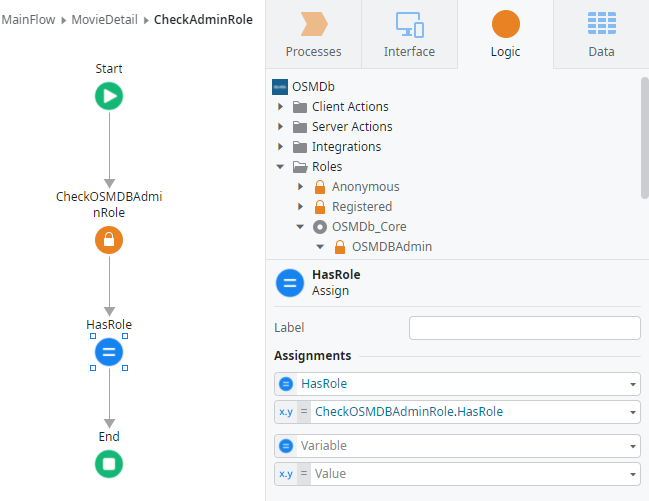


* 1. Abra a Data Action e troque para a aba Logic. Dentro de **Roles->OSMDb\_Core,** você vai achar OSMDbAdmin role. Arraste a **CheckOSMDbAdminRole** e solte no fluxo da Action.

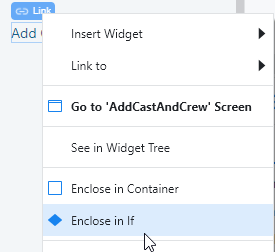


* 1. Arraste um Assign e solte abaixo da Check Action. Defina a atribuição a seguir:

*HasRole = CheckOSMDBAdminRole.HasRole*



* 1. Abra a tela MovieDetail clique com o botão direito em **Add Cast and Crew** e selecione **Enclose in If**

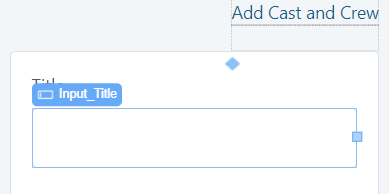


* 1. Defina a **Condition** do If para *CheckAdminRole.HasRole*
  2. Publish o módulo e teste no navegador. Tente acessar anonimamente o AddCastAndCrew. Quando você não conseguir entrar, troque para a conta do usuário OSMDbAdmin e tente novamente.
  3. Aplique a mesma estratégia nos links **New Movie** e **New Person** nas telas **Movies** e **People**.

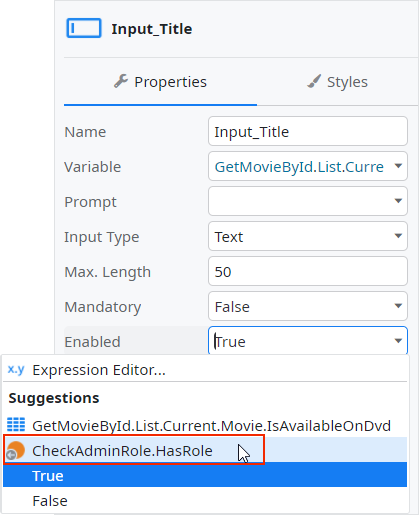
## Inativar Entradas nas Telas

Nessa seção, queremos nos assegurar que a informação sobre filmes e pessoas poderão ser editadas somente pelos usuários OSMDbAdmin. Os outros usuários somente podem ver a informação. Para isso, vamos desativar os campos de entrada.

1. Na tela MovieDetail, se assegure que somente os usuários OSMDbAdmin poderão editar as informações sobre os filmes. Para isso, precisamos desativar as entradas e esconder o save button.
   1. Abra a tela **MovieDetail** e selecione o campo de entrada para o título do filme.

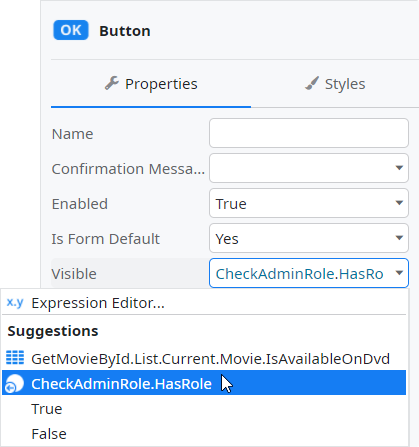


* 1. Nas propriedades da entrada, mude a propriedade **Enabled** (permitido) para *CheckAdminRole.HasRole*

**

**NOTA:** Essa propriedade define em que circunstâncias o campo de entrada tem permissão para ser editado. Quando definido para True, sempre será permitida a edição. Nesse caso, ele só será permitido para usuários com a role OSMDbAdmin.

* 1. Repita esse procedimento para todos os campos remanescentes.
  2. Selecione o botão e defina sua propriedade **Visible** para *CheckAdminRole.HasRole*



* 1. **Publish** e teste as novas funcionalidades.

1. Aplique a mesma estratégia para a tela **PersonDetail**.