

## 21. Navegação Condicional no JSF

Ao contrário da **navegação implícita** — onde o redirecionamento ocorre automaticamente com base no nome da página — a **navegação condicional** é definida por meio de regras explícitas no arquivo de configuração `faces-config.xml`.

Essa forma de navegação oferece **maior controle**, pois permite definir condições lógicas para direcionar o usuário a diferentes páginas, com base no resultado de uma ação ou estado de um bean gerenciável.

### Funcionamento

As regras são escritas dentro das tags `<navigation-rule>` e `<navigation-case>`. Cada `<navigation-rule>` define:

- A página de origem (`<from-view-id>`)
- Um ou mais casos de navegação (`<navigation-case>`)

Os `<navigation-case>` são processados **na ordem em que aparecem**. A **primeira condição satisfeita** será usada, e o fluxo de navegação seguirá para a página definida em `<to-view-id>`.

É possível usar a tag `<if>` (disponível a partir do JSF 2.0) para adicionar uma **verificação condicional** baseada em expressões EL (Expression Language).

### Exemplo:

```
<navigation-rule>
  <from-view-id>/confirmacao.xhtml</from-view-id>

  <navigation-case>
    <from-outcome>index</from-outcome>
    <if>#{clienteBean.novoCadastro}</if>
    <to-view-id>/cadastro.xhtml</to-view-id>
  </navigation-case>

  <navigation-case>
    <from-outcome>index</from-outcome>
    <to-view-id>/index.xhtml</to-view-id>
  </navigation-case>
</navigation-rule>
```

Nesse exemplo:

- Se o método `clienteBean.novoCadastro` retornar `true`, o usuário será redirecionado para `cadastro.xhtml`.
  - Caso contrário, será redirecionado para `index.xhtml`.
- 

## 21.1. Elementos do JSF

O JSF oferece um conjunto robusto de funcionalidades que o tornam um poderoso framework para o desenvolvimento de aplicações web baseadas em componentes. Abaixo estão seus principais elementos:

### Estado dos Componentes

Mantém o **estado dos componentes da UI** entre requisições HTTP, permitindo páginas dinâmicas e interativas.

### Renderização de Componentes

JSF adapta a renderização dos componentes para diferentes **plataformas e navegadores**, garantindo compatibilidade.

### Processamento de Formulários

Gerencia o processamento de um ou mais formulários em uma página, inclusive em **sub-views**.

### Validação

Permite aplicar **validações automáticas** em campos de formulário, com suporte a mensagens de erro customizadas.

### Internacionalização (i18n)

Suporte nativo à **internacionalização**, possibilitando adaptar o sistema a diferentes idiomas e regiões.

### Componentes Visuais

JSF vem com **componentes padrão**, e permite extensão via bibliotecas de terceiros como PrimeFaces, RichFaces etc.

### Árvore de Componentes

A UI é representada como uma **árvore hierárquica de componentes**. Cada componente tem referências ao seu **pai e filhos**.

## **Facets**

Permite adicionar componentes em **posições específicas** e independentes da ordem natural da árvore. Muito usado, por exemplo, em tabelas para definir cabeçalho (**header**) ou rodapé (**footer**).

## **Eventos**

JSF permite a configuração de **listeners** para reagir a eventos como alterações de valores de campos (**ValueChangeEvent**) ou ações de botões (**ActionEvent**).