

Grails

Controllers e Views

Aula 4

Agenda

- Controllers:

- Composição das URLs
- Principais funções
- Objetos implícitos
- Escopos

- Views:

- Expressões
- Tags:
 - Básicas
 - Links
 - Formulários
- Layouts
- Partial

Controllers - URLs

`http://servidor/controlador/ação/id`

Principais Funções

- render
- redirect

Render

- render conteúdo
- render opções
 - text
 - view
 - model
 - contentType

Exemplos

- render “Hello World !”
- render(view: “create”, model: [pessoa: new Pessoa()])

Conversores

- Conversores automáticos para XML e JSON
- É preciso importar este pacote:
 - `import grails.converters.*`
- Exemplo:
 - `render Produto.list() as JSON`

Redirect

- `redirect(opcoes)`
 - `action`
 - `controller`
 - `id`
 - `params`
 - `uri`
 - `url`

Exemplos

- `redirect(action:“show”, id: 4)`
- `redirect(controller:“produto”, action:“create”)`
- `redirect(url:“http://www.google.com”)`

Objetos Implícitos

- params
- flash
- request
- response
- session

Params

- Dicionário contendo todos os dados enviados do cliente
- Exemplo:
 - `params.id`
 - `params.nome`
 - `params.produto`

Response

- Objeto contendo as informações que serão enviadas como resposta para o cliente
- Exemplo:

```
response.getOutputStream << fotografia
```

```
response.getOutputStream.flush( )
```

Escopos

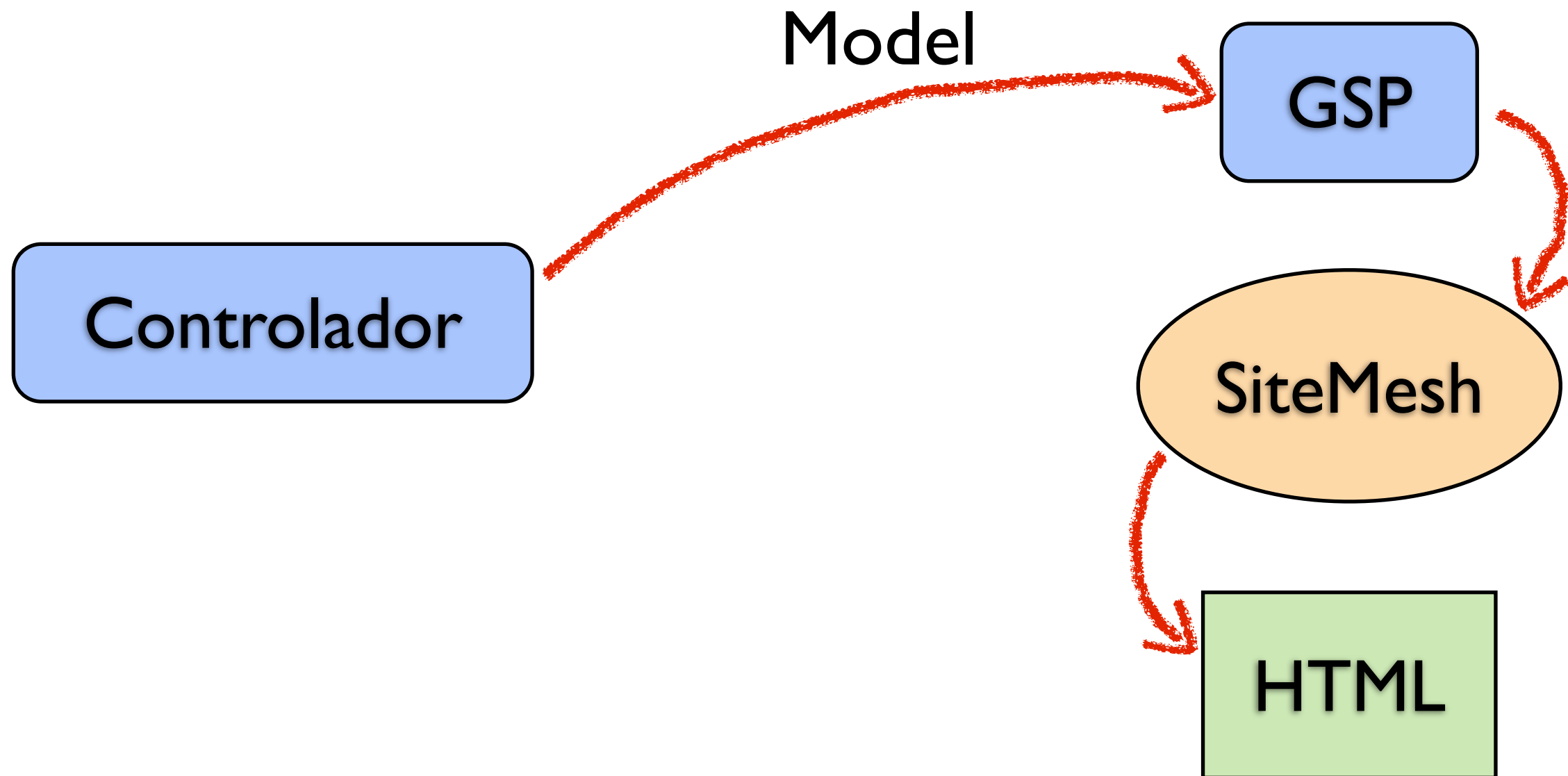
- Dicionários que guardam informações que podem ser armazenadas e/ou recuperadas de acordo com um tempo de vida.

Escopo	Tempo de Vida
request	1 requisição
session	sessão de usuário
flash	2 requisições

Views

- As ações dos controladores são mapeadas para arquivos gsp de acordo com a seguinte estrutura:
- `grails-app/views/controlador/ação.gsp`

Como funciona ?



Expressões

- `${ expressão Groovy }`
- Exemplo:
 - `${ consumidor.nome }`
 - `${ new Date().format('dd/MM/yyyy') }`

Tags Básicas

```
<g:if test="${produto.estoque == 0}">
```

Produto fora de estoque

```
</g:if>
```

```
<g:else>
```

Temos \${produto.estoque} unidades

```
</g:else>
```

Tags Básicas

<g:each var="produto" in="{produtos}">

\${produto.nome}

</g:each>

Tags de Links

```
<g:link controller="post" action="create">Criar Post</g:link>
```

```
" />
```

```
${resource(dir:'css',file:'index.css')}
```

```

```

Tags de Forms

- g:form
- g:uploadForm
- g:textField
- g:passwordField
- g:textArea
- g:select
- g:checkBox
- g:radioGroup
- g:actionSubmit

Tags de Erros

```
<g:hasErrors bean="{empresalInstance}">
```

```
<div class="errors">
```

```
<g:renderErrors bean="{empresalInstance}" as="list" />
```

```
</div>
```

```
</g:hasErrors>
```

Layouts

- Definidos na pasta:
 - `grails-app/views/layouts`
- Referenciados no head das páginas GSP:
 - `<meta name="layout" content="main" />`

Tags de Layout

- `<g:layoutHead />`
- `<g:layoutBody/>`
- `<g:pageProperty name="page.contentHeader"/>`

Tags para Usar Layouts

- `<meta name="layout" content="main" />`
- `<content tag="contentHeader">`

Partials

- Páginas GSP com o prefixo (underline)
- Referenciadas com a tag:
 - `<g:render template="form"/>`
- Parâmetros:
 - bean
 - collection, var
 - model

```
<p>${it.title}</p>  
<p>${it.author}</p>
```

```
<g:render template="displaybook" bean="${book}" />
```

```
<g:render template="displaybook" collection="${books}" />
```