

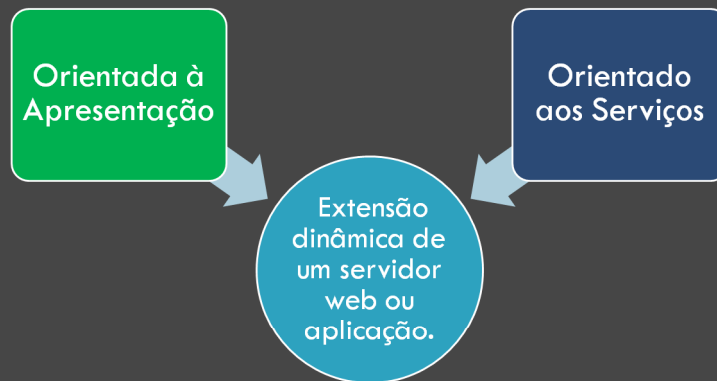
# Tecnologia Java Server Pages

Adolfo Peixinho n.º m4067

Nuno Reis n.º m3955

Sistemas Distribuídos e Tolerância a Falhas 2010/2011

## O que é uma aplicação Web ?

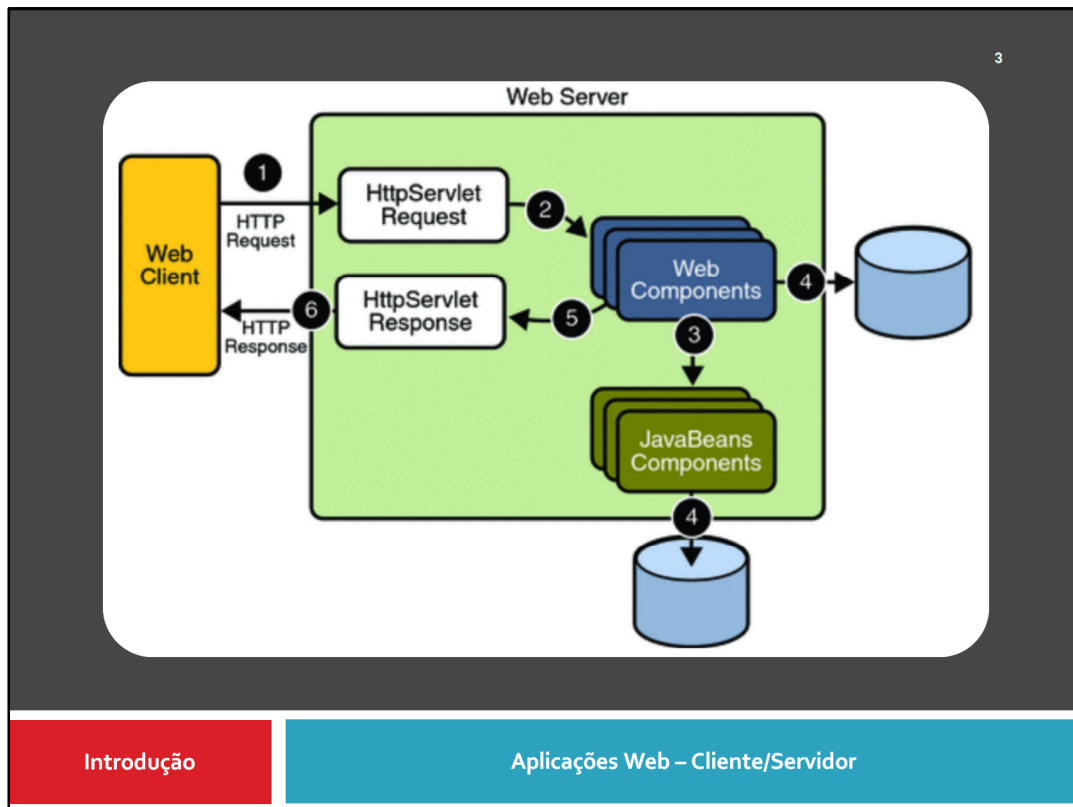


Introdução

O que é uma aplicação Web ?

Orientada a apresentação: uma aplicação web de apresentação orientada gera páginas web interativas com vários tipos de linguagem de marcação (HTML, XML, e assim por diante) e conteúdo dinâmico em resposta a pedidos (Java Servlet)

Orientada a serviços: JAX-WS SOAP API



Os componentes da Web fornecem as capacidades de extensão dinâmica de um servidor web. **Componentes da Web** são ou **servlets**, páginas **JSP**.

Um servidor Web que implementa o **Java Servlet e JavaServer Pages** converte o pedido num objeto **HttpServletRequest**.

Este objeto é entregue a um componente web, que pode interagir com componentes **JavaBeans** ou um **banco de dados para gerar conteúdo dinâmico**.

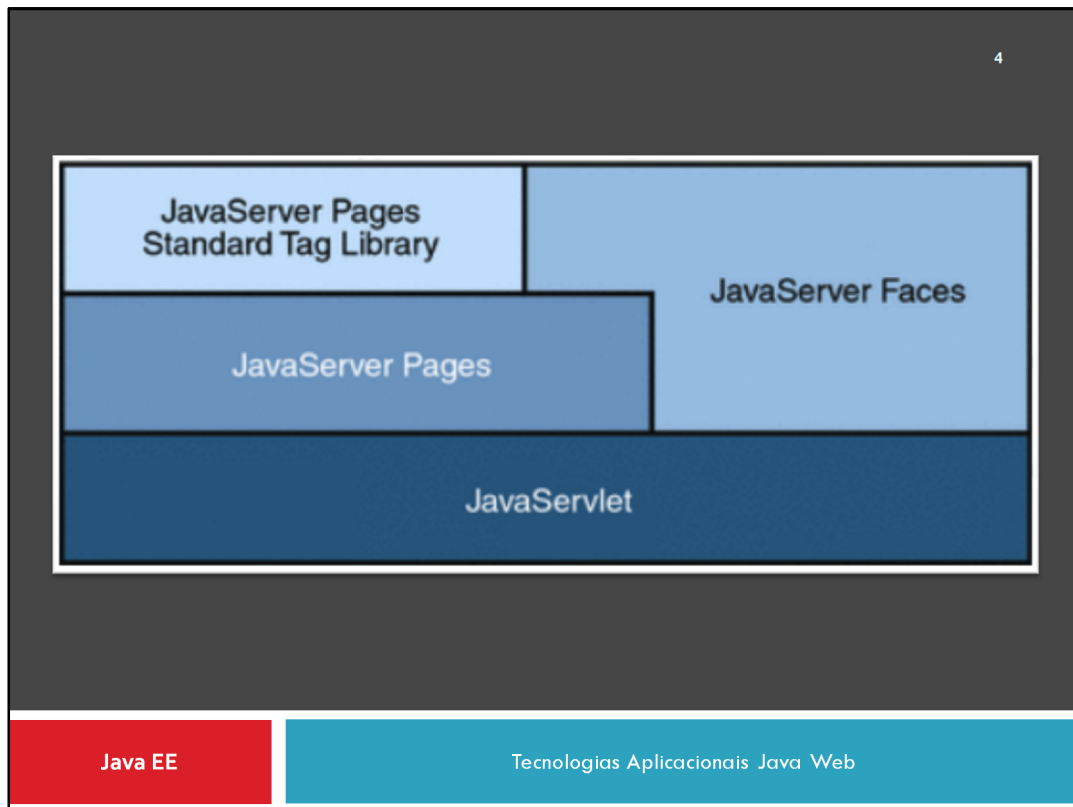
O componente web pode gerar um **HttpServletResponse** ou pode passar a **solicitação para um outro componente web**. Eventualmente, um componente web gera um objeto **HttpServletResponse**. O servidor web converte esse objeto para uma resposta HTTP e devolve para o cliente.

**Servlets** são classes de Java que dinamicamente processam pedidos para construir respostas. **Páginas JSP** são documentos baseados em texto que são executados como servlets, mas permitem uma abordagem mais natural para a criação de conteúdo estático.

Apesar de servlets e páginas JSP podem ser usados alternadamente, cada um tem suas próprias forças.

Servlets são mais adequados para aplicações orientadas a serviços (Web Serviço Endpoints são implementados como servlets) e a funções de controle de uma aplicação orientada a apresentação, tais como envio de pedidos e manipulação de dados não textuais.

Páginas JSP são mais adequadas para a geração de marcação baseada em texto, como HTML, Scalable Vector Graphics (SVG), Wireless Markup Language (WML) e XML.



Desde a introdução da tecnologia Java Servlet e JSP, e outras tecnologias Java e frameworks para a construção de aplicações web interativas têm sido desenvolvidos.

**Observar** que a tecnologia **Java Servlet** é o alicerce de todas as tecnologias de aplicação web.

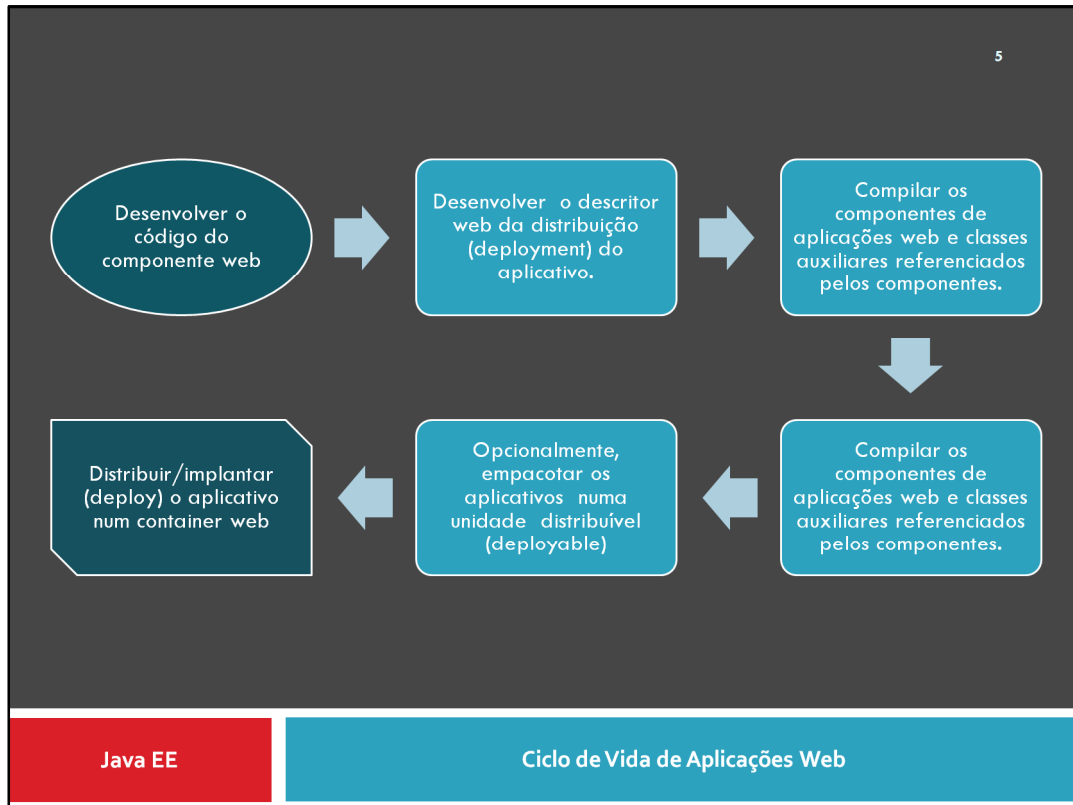
Cada tecnologia adiciona um nível de abstracção que faz prototipagem de aplicações web e de desenvolvimento mais rápido e as aplicações web se mais sustentável, escalável e robusta.

**Componentes Web** são suportados pelos serviços de uma plataforma execução chamado de **container web**.

Um container web oferece serviços como solicitação de expedição, concorrência, segurança e gestão de ciclo de vida.

Dá acesso a APIs de componentes web, tais como nomes, transações, e-mail.

Certos aspectos do comportamento de aplicativo web pode ser configurado quando o aplicativo é instalado, ou implantados, para o container web. As informações de configuração são mantidas em um arquivo de texto no formato XML chamado um descritor de implantação de aplicativos web (DeploymentDescriptor). A DD deve estar em conformidade com o esquema descrito na especificação Java Servlet.

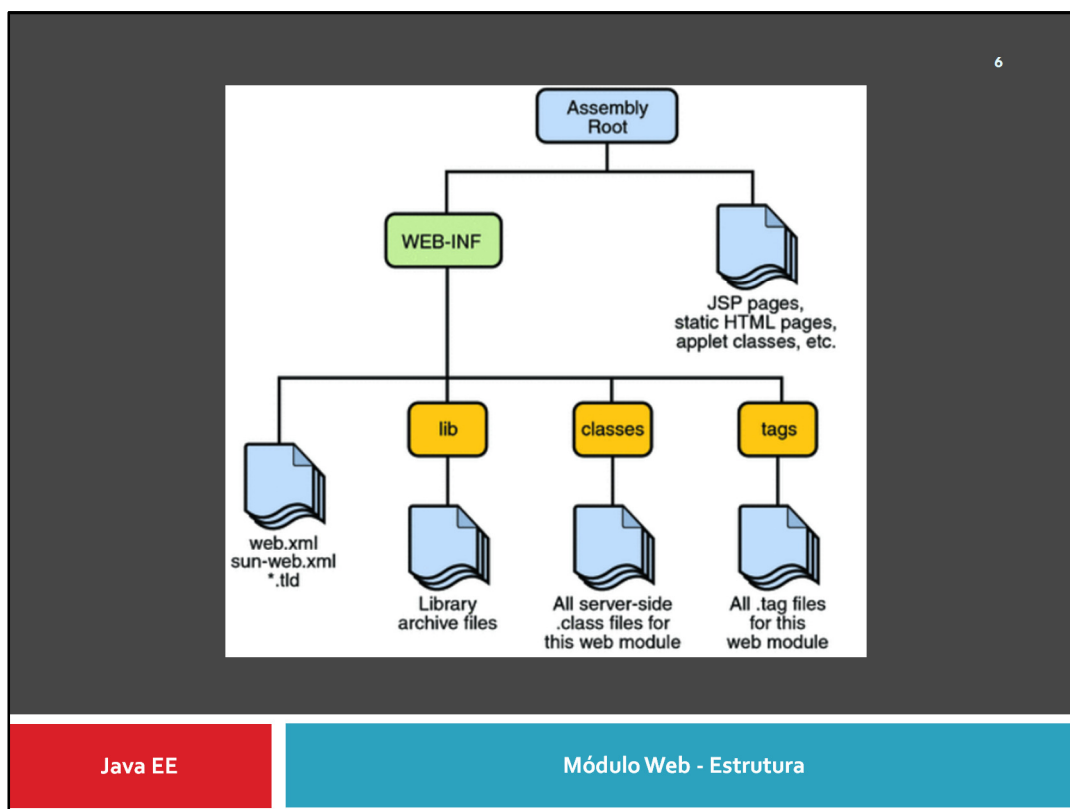


Um aplicativo da Web consiste em **componentes web**, arquivos de recursos estáticos, como imagens e classes auxiliares e bibliotecas.

O container web fornece muitos serviços de apoio que reforcem as capacidades dos componentes web e torná-los mais fáceis de desenvolver.

No entanto, como uma aplicação web deve levar em conta esses serviços, o processo de criação e execução de uma aplicação web é diferente da tradicional classes autônomas Java

1. Desenvolver o código do componente web.
2. Desenvolver a web descritor de implementação do aplicativo.
3. Compilar os componentes de aplicações web e classes auxiliares referenciados pelos componentes.
4. Opcionalmente, o pacote de aplicativos em uma unidade implantável.
5. Implantar (DEPLOY) o aplicativo em um container web.
6. Aceder uma URL que faz referência a aplicação web.



Java EE

Módulo Web - Estrutura

**Componentes Web** e arquivos de conteúdo estático da Web, como as imagens são chamados de recursos web.

Um **módulo web** é a menor unidade deployable/distribuível e utilizável dos recursos da web.

Um **módulo Java EE Web** corresponde a uma aplicação web, tal como definido na especificação Java Servlet

Além de componentes web e recursos da Web, um módulo web pode conter outros arquivos:

Do lado do **servidor** classes utilitárias (BeansDB JavaBeans). Muitas vezes, essas classes estão em conformidade com a arquitetura de componentes JavaBeans.

Do lado do **cliente** classes (applets e classes de serviço).

Um módulo Web tem uma estrutura específica. O diretório de nível superior de um módulo web é a raiz de pedidos de documentos.

A raiz do documento são armazenadas as páginas JSP, classes client-side e arquivos, recursos da Web estáticos, como imagens.

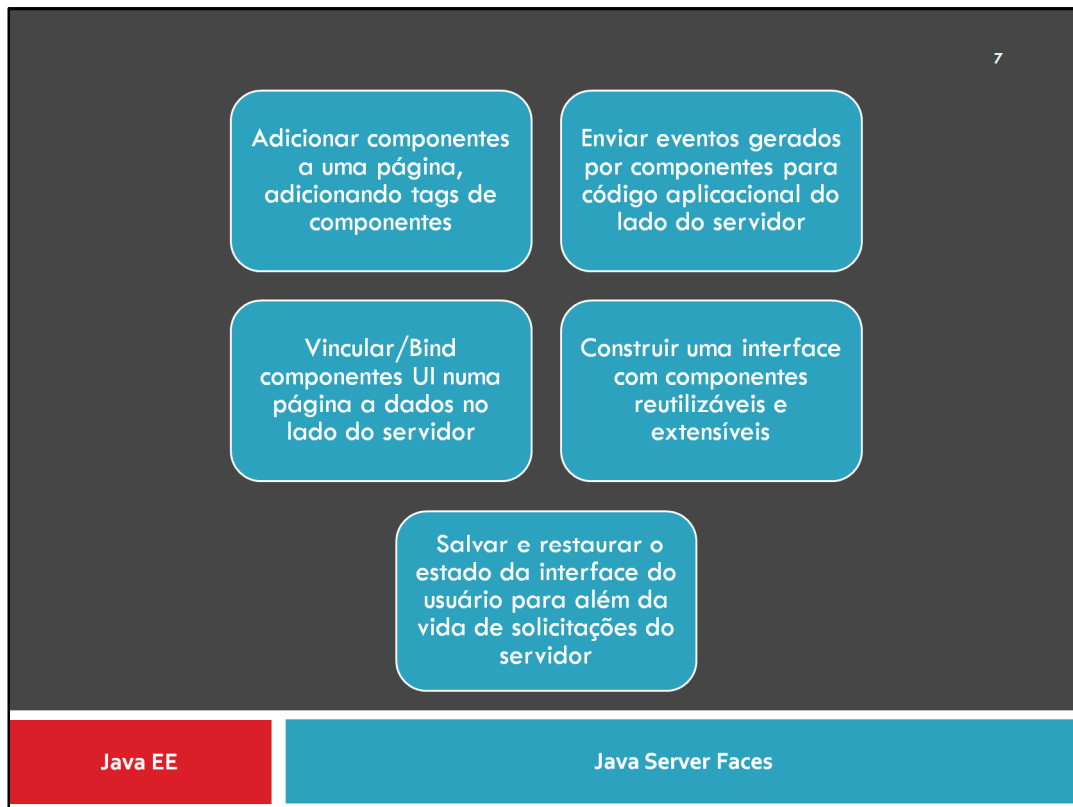
A raiz do documento contém um subdiretório chamado WEB-INF, que contém os seguintes arquivos e diretórios:

web.xml: O descritor de distribuição/deployment de aplicativos web

biblioteca de arquivos do descritor Tag (ver Descritores Tag Library)

classes: um directório que contém no lado do servidor classes: servlets, classes de utilitários e componentes JavaBeans

tags: Uma pasta que contém arquivos de tag, que são implementações de bibliotecas



JavaServer Faces é uma tecnologia do lado do servidor componente de interface da framework de usuário para aplicações web baseadas na tecnologia Java. Uma API para representar componentes UI e gerenciar seu estado; manipulação de eventos, validação do lado do servidor, conversão de dados, definição de navegação de página; apoio internacionalização e acessibilidade; e extensibilidade, que prevê todas essas características

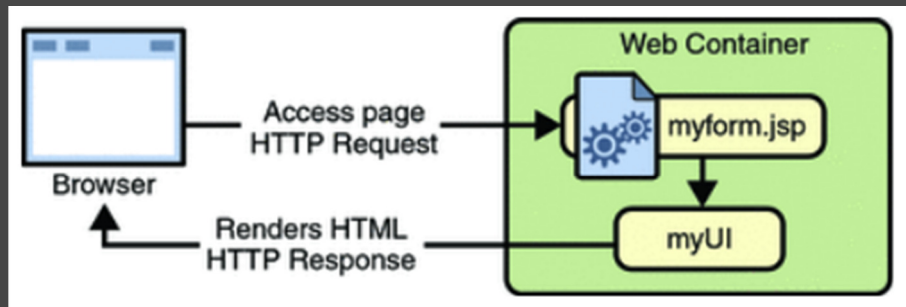
Duas bibliotecas de tags personalizadas para expressar os componentes de interface do usuário dentro de uma página JSP e de componentes para a ligação de objetos do lado do servidor.

**h** This tag library contains JavaServer Faces component tags for all UIComponent + HTML RenderKit Renderer combinations defined in the JavaServer Faces Specification.

**f** The core JavaServer Faces custom actions that are independent of any particular RenderKit

Vantagens JSF O modelo de programação bem definida e bibliotecas de tag aliviar significativamente o fardo da construção e manutenção de aplicações web com interfaces de usuário do servidor.

Separação entre comportamento e apresentação



○ UI criado com JavaServer Faces (myUI) executa no servidor e devolvido ao cliente.

Java EE

Java Server Faces

A pagina JSP, myform.jsp, que inclui JavaServer Faces tags.

Expressa componentes UI usando tags definidas por JSF.

Any event listeners, validators, and converters that are registered on the components

The JavaBeans components that encapsulate the data and application-specific functionality of the components



Um conjunto de páginas JSP (embora não limitado a páginas JSP como tecnologia de apresentação)

Um conjunto de componentes JavaBeans que definem propriedades e funções para UI numa página

Um aplicativo de configuração do arquivo de recursos, que define regras de navegação de página e configura Beans e outros objectos personalizados, tais como componentes

Um descritor de distribuição/deployment (um arquivo web.xml)

Objectos personalizados criados pelo desenvolvedor do aplicativo. Estes objectos podem incluir componentes personalizados, validadores, conversores, ou listeners.

Um conjunto de tags personalizadas para representar objectos personalizados na página

Java EE

Java Server Faces

# Exemplo de Aplicação Web - Netbeans

Java EE

Java Server Faces – Exemplo Netbeans

Questões ???

Obrigado...