



Workshop:  
Crea un CRUD completo en **JSF**  
integrando **Primefaces** y  
**Spring Boot** usando **API REST**

**Jean Ramal**  
Expositor



# Agenda

## **Workshop: Crea un CRUD completo en JSF integrando Primefaces y Spring Boot usando API REST**

- ¿Qué es Jakarta Server Faces (JSF), Primefaces, Spring Boot y API REST?.
- Breve introducción a JSF, Primefaces, Spring Boot y API REST.
- Implementar CRUD en JSF con Primefaces, Spring Boot y API REST.

# Tecnologías a usar

1. JDK
2. Tomcat
3. IDE Spring Tool Suite / Eclipse
4. JSF
5. PrimeFaces
6. Spring Boot
7. Lombok
1. Postman \*
8. Maven
1. Spring initializr

# Jakarta Server Faces (JSF)

## ¿Que es Jakarta Server Faces?

- Es un framework MVC para crear interfaces de usuario para aplicaciones web, incluyendo los componentes, gestión de estado, eventos, validaciones, navegación y el soporte para la internacionalización y la accesibilidad.
- Proporciona un modelo simple para conectar eventos del cliente al código del lado del servidor.
- Está diseñado para aliviar significativamente el desarrollo y mantenimiento de sistemas web.
- Permite que los componentes personalizados de UI se construyan y se reutilicen fácilmente.
- Establece estándares para ser aprovechados por desarrolladores y herramientas.

### Últimas versiones

JSF 2.2 (16-04-2013) para JEE 7 – HTML5, Faces Flow, Stateless views.

JSF 2.3 (28-03-2017) para JEE 8– mayor soporte CDI, WebSockets, URL sin extensión. URL: <https://javaee.github.io/javaxserverfaces-spec/>

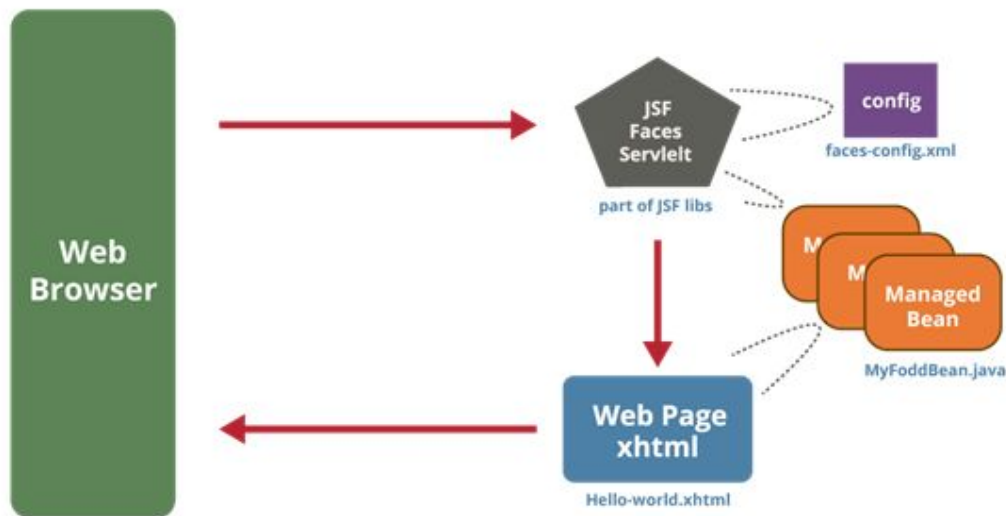
JSF 3.0 (06-05-2020) para JEE 9. URL: <https://jakarta.ee/specifications/faces/>



**JAKARTA™ EE**

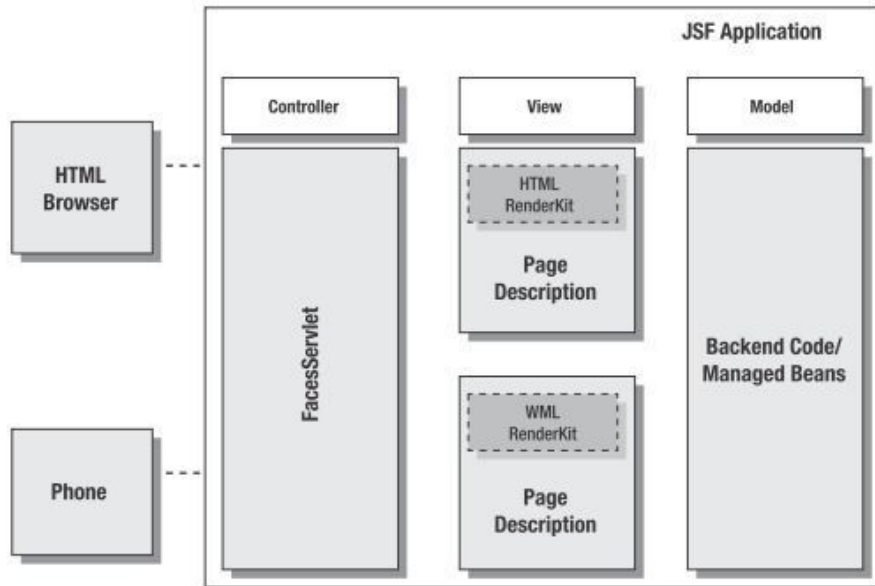
# Jakarta Server Faces (JSF)

## Componentes Principales



# Jakarta Server Faces (JSF)

## Patrón: Model View Controller (MVC)



# Jakarta Server Faces (JSF)

## Managed Bean

- Son clases Java gestionados por el marco JSF que contienen sus métodos de acceso (getters and setters), lógica de negocio o incluso otras clases de respaldo.
- Funcionan como modelo para los componentes UI.
- Sus funciones más comunes son: validación de datos, manejo de eventos disparado por un componente, procesamiento de navegación.

```
@Named
@RequestScoped
public class AutoCompleteController {

    private String txt1;

    public List<String> completeText(String query) {
        List<String> results = new ArrayList<>();
        for(int i = 0; i < 10; i++) {
            results.add(query + " " + i);
        }

        return results;
    }

    public String getTxt1() {
        return txt1;
    }

    public void setTxt1(String txt1) {
        this.txt1 = txt1;
    }

}
```



# Jakarta Server Faces (JSF)

## Configuración de Managed Beans

En JSF 1.2 un bean administrado tenía que registrarse en el archivo **faces-config.xml**. Desde JSF 2.0 en adelante, se pueden registrar usando anotaciones.

### Usando la configuración XML

```
<managed-bean>
  <managed-bean-name>holaMundo</managed-bean-name>
  <managed-bean-class> pe.com.dominio.training.test.HolaMundo </managed-bean-class>
  <managed-bean-scope>request</managed-bean-scope>
</managed-bean>
```

### Java CDI

### Usando la anotación

```
import javax.faces.bean.RequestScoped;
import javax.faces.bean.ManagedBean;

@ManagedBean
@RequestScoped
public class HolaMundo {

}
```

```
import javax.inject.Named;
import javax.enterprise.context.RequestScoped;

@Named
@RequestScoped
public class HolaMundo {

}
```

# Jakarta Server Faces (JSF)

## Scope de Managed Beans

Existen diferentes formas de almacenar temporalmente la información.  
Se listan los principales:

### Request Scoped

El bean vive mientras la solicitud HTTP viva.

### View Scoped

El bean vive mientras el usuario interactúe con la misma vista.

### Session Scoped

El bean vive mientras dure la sesión HTTP.

### Application Scoped

El bean vive mientras dura la aplicación web.



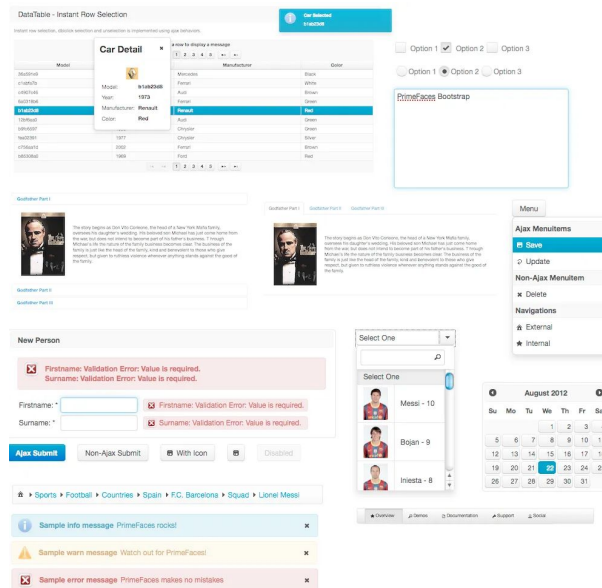
# PrimeFaces

PrimeFaces es una biblioteca de código abierto y muy popular para JSF. Es la suite de UI más completa del mercado.

## ¿Por qué PrimeFaces?

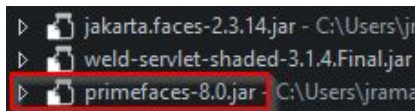
- Más de 100 componentes
- Simplicidad y rendimiento
- Facilidad de uso
- Comunidad activa
- Flexible
- Varios temas gratuitos y Premium
- Soporte html5
- Diseño responsivo
- Es el framework JSF más usado en el mundo

<https://www.primefaces.org/>



# PrimeFaces - Get Started

PrimeFaces es una biblioteca ligera con un **jar**, configuración cero y sin dependencias requeridas. Solo necesita descargar PrimeFaces, agregar el **jar** a su classpath y colocar el **namespace** para comenzar.



Un JAR



```
<?DOCTYPE html>
<html xmlns="http://www.w3.org/1999/xhtml"
      xmlns:h="http://xmlns.jcp.org/jsf/html"
      xmlns:p="http://primefaces.org/ui">
</html>

<h:head>
  <title>Componentes Primefaces</title>
  <h:outputStylesheet name="css/style.css" />
</h:head>
<h:body>
  <h1>Spinner</h1>
  <p:spinner />
</h:body>
</html>
```

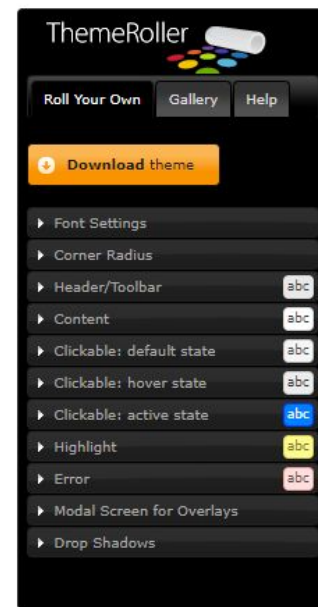
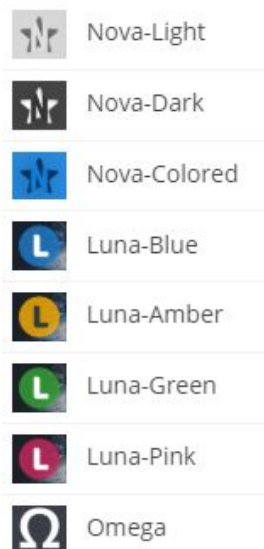
Namespace y componente



Componente HTML con estilo

<https://github.com/primefaces/primefaces>

# Temas Free



Nuevos temas free

Legados (free)  
no recomendables para versiones actuales de PF

Diseñador Legado

# Temas Premium



Temas y Layouts Premium

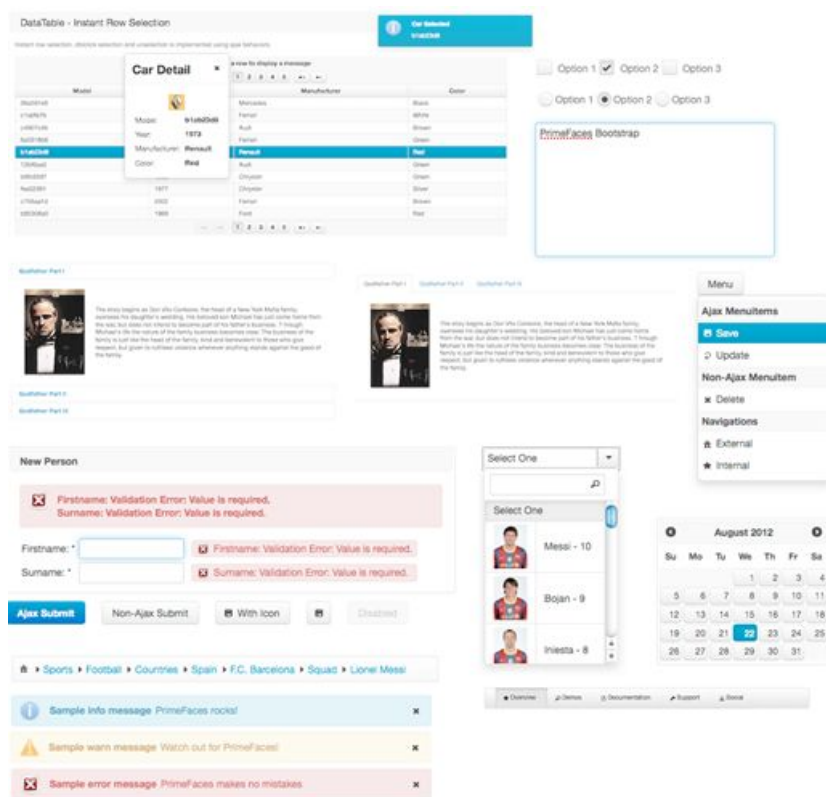


Diseñador

# Componentes

Más de 100 componentes de distintos tipos:

- Ajax Core
- Inputs
- Buttons
- Data
- Panels
- Overlays
- Menus
- Charts (Gráficos)
- Messages
- Multimedia
- Files
- Drag and Drop
- Cliente Validation
- Dialog Framework
- Misc



<https://www.primefaces.org/showcase/>

# PrimeFaces Extensions

Es una biblioteca ligera de código abierto para JSF, impulsado por la comunidad. El proyecto se base en Primefaces y tiene componentes extendidos faltantes en otras bibliotecas JSF o componentes mejorados.

- Más de 30 componentes
- Conversores
- Utilidades
- Componentes incluidos en Primefaces



<http://primefaces-extensions.github.io/>

<https://www.primefaces.org/showcase-ext/views/home.jsf>

<https://github.com/primefaces-extensions/primefaces-extensions.github.com>



# REST (**R**epresentational **S**tate **T**ransfer)

## ¿Qué es REST?

Es un estilo de arquitectura que define unos principios y técnicas para el desarrollo de software que permite describir cualquier interface web y usa la especificación HTTP.

Conceptos principales:

- **Recursos:** /usuario
- **Verbos HTTP:** GET, POST, PUT, DELETE, otros
- **Código de Estado:** 404 (not found)
- **Controles de Hipermedia:** Links en una paginación

\*Modelo de madurez de Richardson

# Spring Framework

## ¿Por qué Spring?

Spring hace que la programación de Java sea más rápida, fácil y segura para todos. El enfoque de Spring en la velocidad, la simplicidad y la productividad lo han convertido en el framework Java más popular del mundo .

Podemos trabajar con Microservicios, de forma Reactiva, Nube, Web, Serverless, Conducido por evento y Batch.

Más de 20 proyectos para implementar.



# Spring Framework

## ¿Qué es Spring Boot?

- Es un proyecto de Spring que sirve como base para todos los demás proyectos, para la creación del software.
- Facilita la creación de aplicaciones stand-alone listos para producción que puede ejecutarlo fácilmente.
- Configuración automática de librerías de Spring y de terceros siempre que sea posible.
- Ninguna configuración XML.



¿Tienen alguna pregunta?

# ¡GRACIAS!



IG: @perfecci0ndigital

FB: @perfecci0ndigital