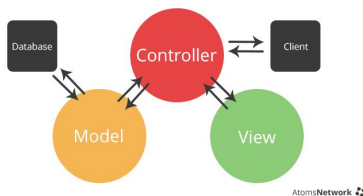
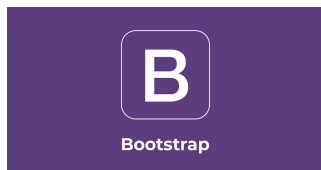


# STRUTS 2

UNIVERSIDAD DE LA SIERRA SUR

# Puntales Apache

Apache Struts es un marco MVC gratuito y de código abierto para crear aplicaciones web Java elegantes y modernas. Prefiere la convención sobre la configuración, es extensible mediante una arquitectura de complementos y se envía con complementos para admitir REST, AJAX y JSON.



# Tecnologías clave

- ❑ HTTP y HTML
- ❑ El ciclo de solicitud/respuesta HTTP
- ❑ El lenguaje Java y los marcos de aplicación
- ❑ JavaScript, AJAX y SOAP
- ❑ Archivos de propiedades y paquetes de recursos
- ❑ Servlets, Filtros y Contenedores Web
- ❑ Sesiones
- ❑ JavaServer Pages y bibliotecas de etiquetas JSP
- ❑ Lenguaje de marcado extensible (XML)
- ❑ JAAS
- ❑ Controlador de vista de modelo

# MySQL - Docker

1. Descargar la imagen de mysql

```
docker pull mysql
```

2. Crear el contenedor de mysql

```
docker run --name mysqlserver -e MYSQL_ROOT_PASSWORD=root -d mysql:latest
```

3. Entrar al contenedor de mysql

```
docker exec -it <idcontenedor o nombrecontenedor> /bin/bash
```

4. Iniciar en línea de comandos de mysql

```
mysql -u root -ppassword o mysql -u root -p
```

5. Ingresar la contraseña

## Workbench

<https://dev.mysql.com/downloads/workbench/>

# Comandos docker

1. Iniciar el contenedor

**docker start iddelcontenedor**

2. Detener el contenedor

**docker stop iddelcontenedor**

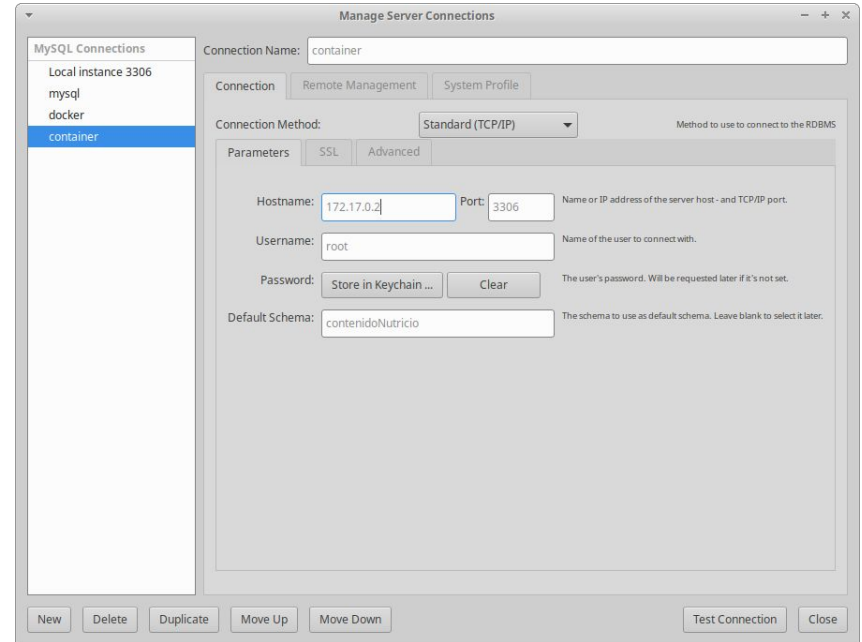
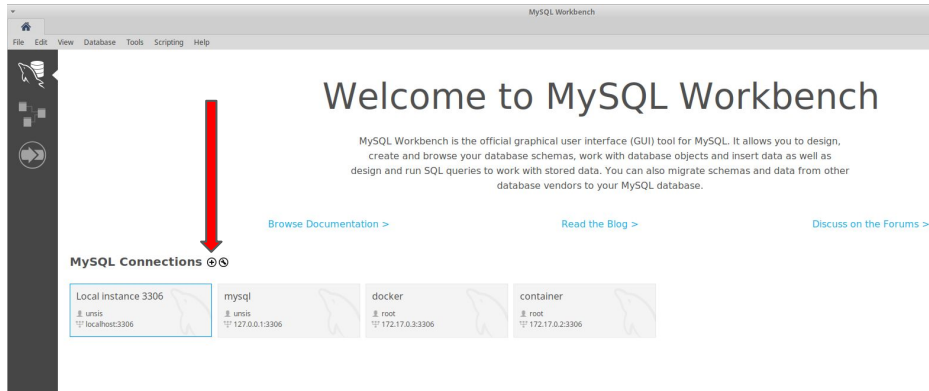
3. Listar contenedores activos e inactivos

**docker ps -a**

4. Revisar propiedades (ip, puerto, etc) del contenedor

**docker inspect iddelcontenedor**

# Configuración de workbench y MySQL



# Construcción de la base de datos

```
CREATE DATABASE IF NOT EXISTS banco;
```

```
DROP DATABASE IF EXISTS banco;
```

```
CREATE DATABASE banco;
```

```
USE banco;
```

```
DROP TABLE IF EXISTS `usuario`;
```

```
CREATE TABLE `usuario` (
```

```
  `idUsuario` int NOT NULL AUTO_INCREMENT,
```

```
  `nombreUsuario` varchar(20) DEFAULT NULL,
```

```
  `contraseña` varchar(20) DEFAULT NULL,
```

```
  `fechaCreacion` Date DEFAULT NULL,
```

```
  `fechaActualizacion` Date DEFAULT NULL,
```

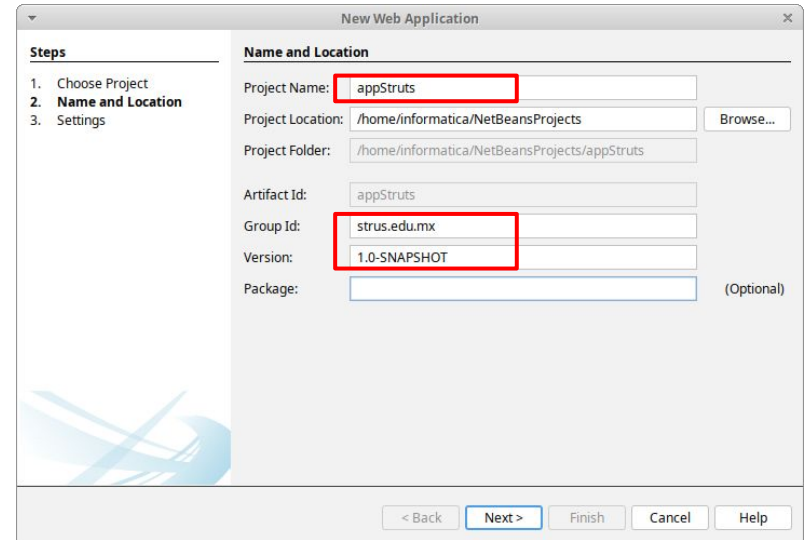
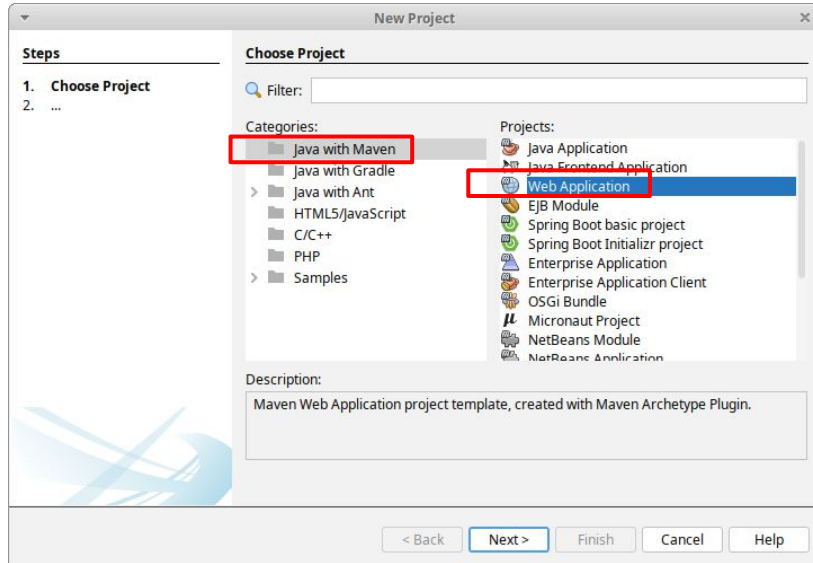
```
  `fechaEliminacion` Date DEFAULT NULL,
```

```
  PRIMARY KEY (`idUsuario`)) ENGINE=InnoDB
```

```
DEFAULT CHARSET=utf8mb4
```

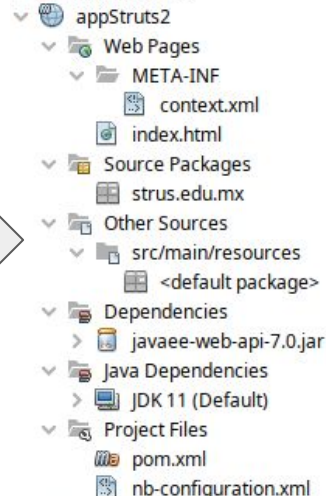
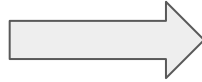
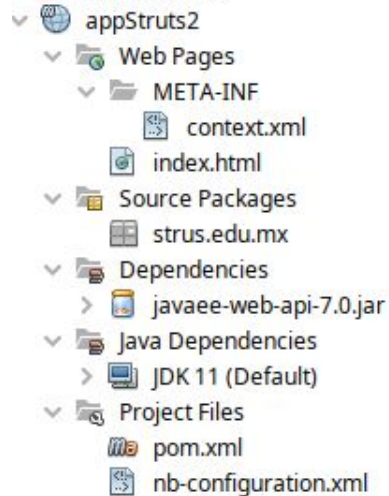
```
COLLATE=utf8mb4_0900_ai_ci;
```

# Construcción del proyecto Struts 2

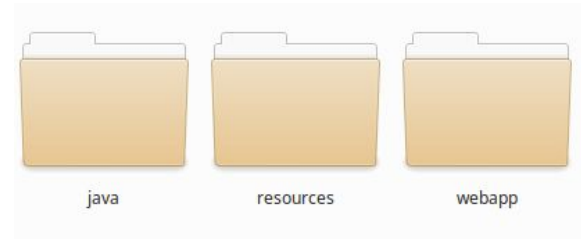




# Construcción del proyecto Struts 2



1. Agregar manualmente el directorio resources ***appStruts2/src/main/***



# Dependencias

```
<!-- https://mvnrepository.com/artifact/org.hibernate/hibernate-core -->
<dependency>
    <groupId>org.hibernate</groupId>
    <artifactId>hibernate-core</artifactId>
    <version>6.1.4.Final</version>
    <type>pom</type>
</dependency>

<!-- https://mvnrepository.com/artifact/mysql/mysql-connector-java -->
<dependency>
    <groupId>mysql</groupId>
    <artifactId>mysql-connector-java</artifactId>
    <version>8.0.28</version>
</dependency>

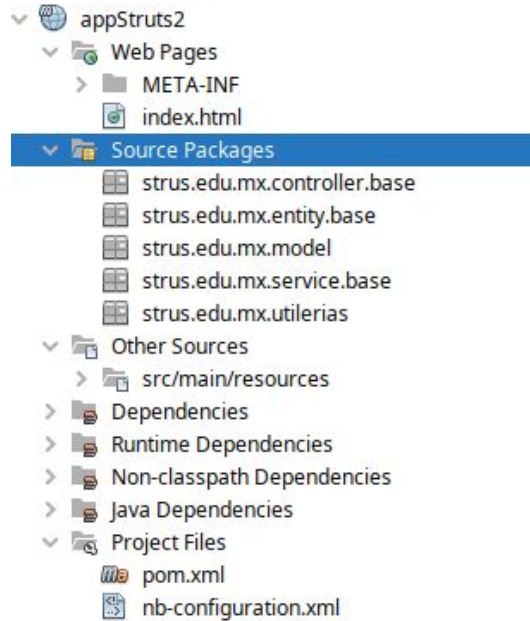
<!-- https://mvnrepository.com/artifact/org.webjars/bootstrap -->
<dependency>
    <groupId>org.webjars</groupId>
    <artifactId>bootstrap</artifactId>
    <version>5.2.2</version>
</dependency>
```

```
<!-- https://mvnrepository.com/artifact/org.webjars.npm/bootstrap-icons -->
<dependency>
    <groupId>org.webjars.npm</groupId>
    <artifactId>bootstrap-icons</artifactId>
    <version>1.10.2</version>
</dependency>

<!-- https://mvnrepository.com/artifact/org.webjars.bower/jquery -->
<dependency>
    <groupId>org.webjars.bower</groupId>
    <artifactId>jquery</artifactId>
    <version>3.6.1</version>
</dependency>
```

# Construcción del proyecto Struts 2

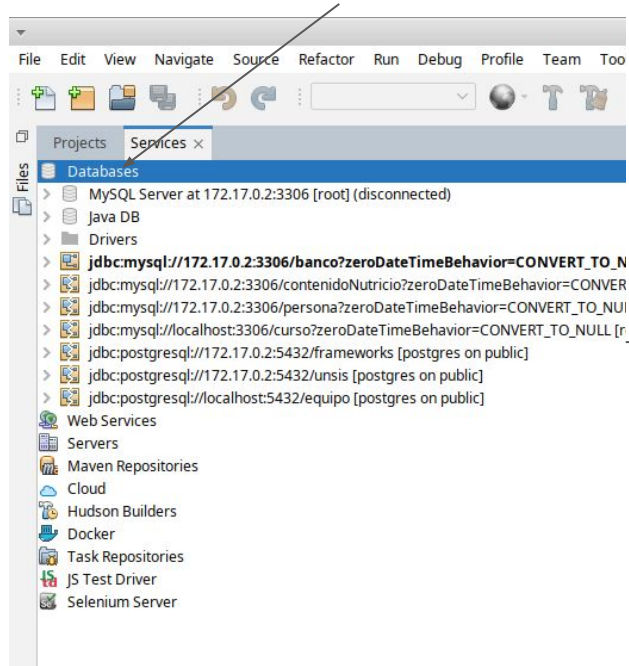
## Creación de paquetes



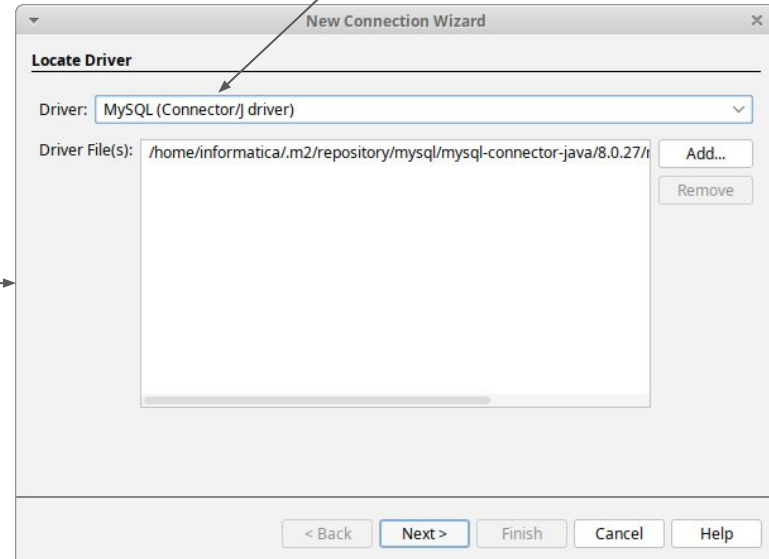
## Archivo de configuración hibernate.cfg.xml

# Construcción del proyecto Struts 2

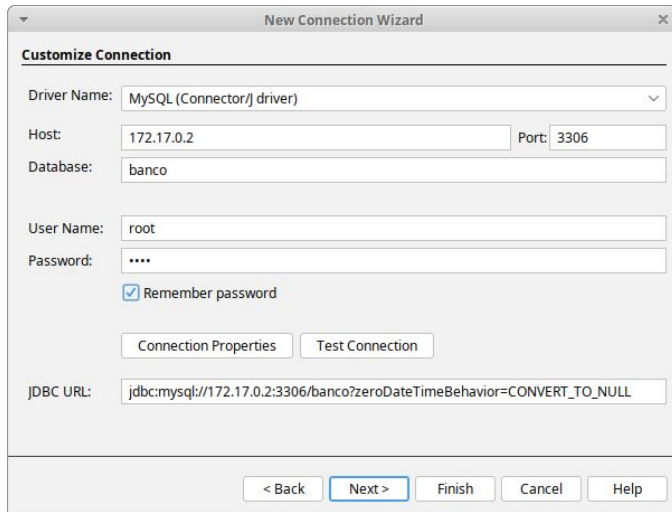
Click derecho -> New connection



Seleccionar el driver



# Construcción del proyecto Struts 2



The 'New Connection Wizard' window, titled 'New Connection Wizard', shows the 'Customize Connection' step. It contains several input fields: 'Driver Name' is a dropdown menu showing 'MySQL (Connector/J driver)'; 'Host' is a text field with '172.17.0.2'; 'Port' is a text field with '3306'; 'Database' is a text field with 'banco'; 'User Name' is a text field with 'root'; 'Password' is a text field with masked characters '....'; and a checked checkbox for 'Remember password'. Below these fields are two buttons: 'Connection Properties' and 'Test Connection'. At the bottom, there is a 'JDBC URL' text field containing 'jdbc:mysql://172.17.0.2:3306/banco?zeroDateTimeBehavior=CONVERT\_TO\_NULL'. The navigation bar at the bottom includes '< Back', 'Next >', 'Finish', 'Cancel', and 'Help' buttons.

**New Connection Wizard**

**Customize Connection**

Driver Name: MySQL (Connector/J driver)

Host: 172.17.0.2 Port: 3306

Database: banco

User Name: root

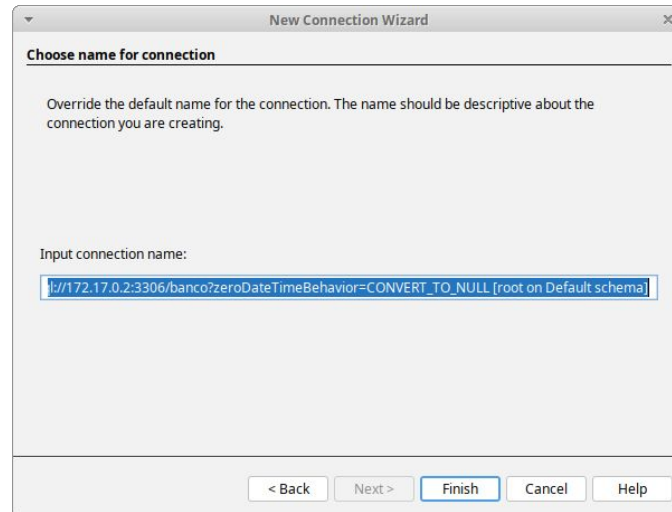
Password: ....

☒ Remember password

Connection Properties Test Connection

JDBC URL: jdbc:mysql://172.17.0.2:3306/banco?zeroDateTimeBehavior=CONVERT\_TO\_NULL

< Back Next > Finish Cancel Help



The 'New Connection Wizard' window, titled 'New Connection Wizard', shows the 'Choose name for connection' step. It contains a text area with the instruction: 'Override the default name for the connection. The name should be descriptive about the connection you are creating.' Below this is a label 'Input connection name:' followed by a text field containing the JDBC URL 'jdbc:mysql://172.17.0.2:3306/banco?zeroDateTimeBehavior=CONVERT\_TO\_NULL [root on Default schema]'. The navigation bar at the bottom includes '< Back', 'Next >', 'Finish', 'Cancel', and 'Help' buttons.

**New Connection Wizard**

**Choose name for connection**

Override the default name for the connection. The name should be descriptive about the connection you are creating.

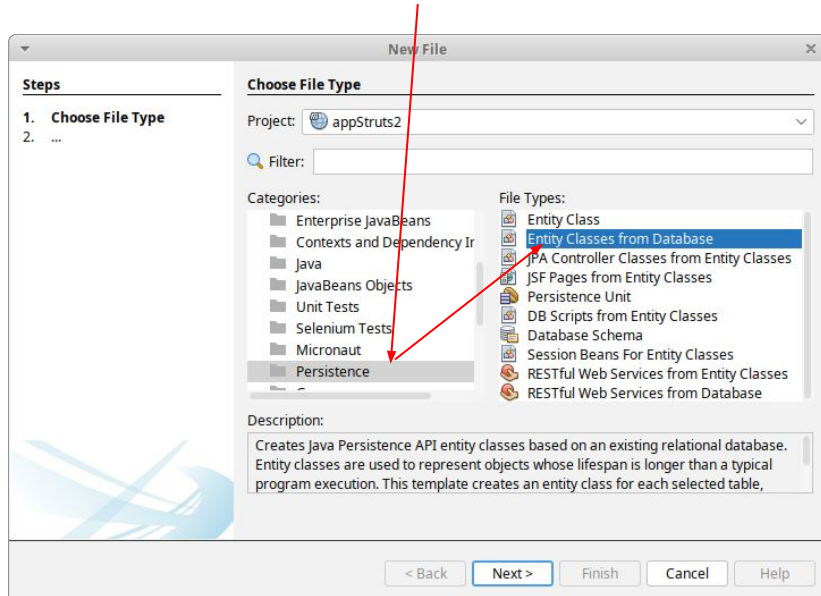
Input connection name:

jdbc:mysql://172.17.0.2:3306/banco?zeroDateTimeBehavior=CONVERT\_TO\_NULL [root on Default schema]

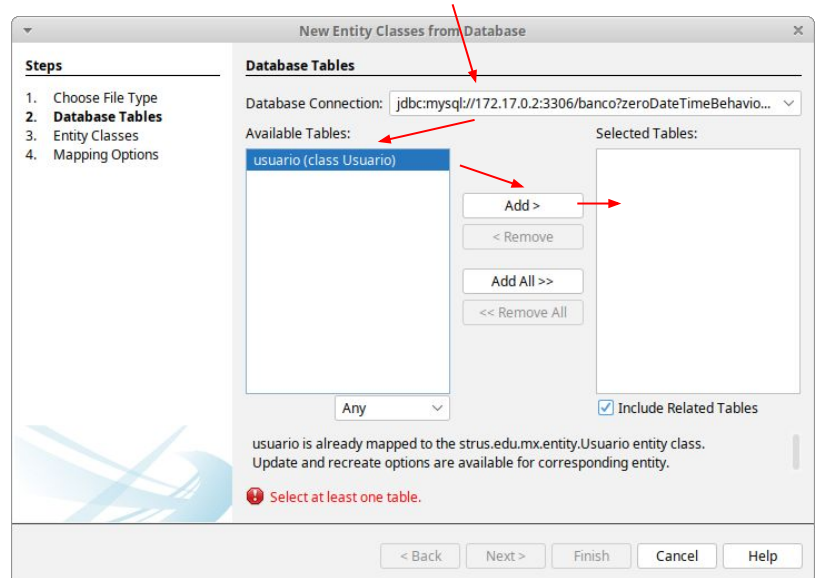
< Back Next > Finish Cancel Help

# Construcción del proyecto Struts 2

Click derecho sobre el proyecto -> New -> Other



Seleccionar la bd y todas las tablas para mapear



# Construcción del proyecto Struts 2

Seleccionar el paquete y propiedades

**New Entity Classes from Database**

**Steps**

1. Choose File Type
2. Database Tables
- 3. Entity Classes**
4. Mapping Options

**Entity Classes**

Specify the names and the location of the entity classes.

Database Table	Class Name	Generation Type
usuario	Usuario	Update

Project: appStruts2

Location: Source Packages

Package: strus.edu.mx.controller.base

☐ Generate Named Query Annotations for Persistent Fields

☐ Generate JAXB Annotations

☐ Generate MappedSuperclasses instead of Entities

< Back **Next >** Finish Cancel Help

**New Entity Classes from Database**

**Steps**

1. Choose File Type
2. Database Tables
3. Entity Classes
- 4. Mapping Options**

**Mapping Options**

Specify the default mapping options.

Association Fetch: default

Collection Type: java.util.Collection

☐ Fully Qualified Database Table Names

☐ Attributes for Regenerating Tables

☒ Use Column Names in Relationships

☐ Use Defaults if Possible

☐ Generate Fields for Unresolved Relationships

< Back Next > **Finish** Cancel Help

# Construcción del proyecto Struts 2

Se genera las siguientes clases (POJO o beans)

```
@Entity
@Table(name = "usuario")
public class Usuario implements Serializable {

    private static final long serialVersionUID = 1L;
    @Id
    @GeneratedValue(strategy = GenerationType.IDENTITY)
    @Basic(optional = false)
    @Column(name = "idUsuario")
    private Integer idUsuario;
    @Size(max = 20)
    @Column(name = "nombreUsuario")
    private String nombreUsuario;
    @Size(max = 20)
    @Column(name = "contrase\u000f1a")
    private String contrase\u00f1a;
    @Column(name = "fechaCreacion")
    @Temporal(TemporalType.DATE)
    private Date fechaCreacion;
    @Column(name = "fechaActualizacion")
    @Temporal(TemporalType.DATE)
    private Date fechaActualizacion;
    @Column(name = "fechaEliminacion")
    @Temporal(TemporalType.DATE)
    private Date fechaEliminacion;

    public Usuario() {
    }

    public Usuario(Integer idUsuario) {
        this.idUsuario = idUsuario;
    }

    public Integer getIdUsuario() {
        return idUsuario;
    }

    public void setIdUsuario(Integer idUsuario) {
        this.idUsuario = idUsuario;
    }

    public String getNombreUsuario() {
        return nombreUsuario;
    }
}
```

## 2. Implementar la interfaz

```
@Entity
@Table(name = "usuario")
public class Usuario implements Serializable, IRegistroEntityBase {

    private static final long serialVersionUID = 1L;
    @Id
    @GeneratedValue(strategy = GenerationType.IDENTITY)
    @Basic(optional = false)
    @Column(name = "idUsuario")
    private Integer idUsuario;
    @Size(max = 20)
    @Column(name = "nombreUsuario")
    private String nombreUsuario;
    @Size(max = 20)
    @Column(name = "contrase\u000f1a")
    private String contrase\u00f1a;
    @Column(name = "fechaCreacion")
    @Temporal(TemporalType.DATE)
    private Date fechaCreacion;
    @Column(name = "fechaActualizacion")
    @Temporal(TemporalType.DATE)
    private Date fechaActualizacion;
    @Column(name = "fechaEliminacion")
    @Temporal(TemporalType.DATE)
    private Date fechaEliminacion;
}
```

## 1. Crear una interfaz para el manejo genérico de los id en el package strus.edu.mx.entity.base

package strus.edu.mx.entity.base;

```
public interface IRegistroEntityBase {
    public int getId();
    public void setId(Integer id);
}
```

## 3. Agregar los métodos

```
@Override
public void getId() {
    return getIdUsuario();
}

@Override
public void setId(Integer id) {
    setIdUsuario(idUsuario);
}
```



# Construcción del proyecto Struts 2 - Capa Model

```
package strus.edu.mx.model;
```

```
import java.util.List;  
import strus.edu.mx.entity.Usuario;
```

```
public interface IUserioModel {  
    public void crearRegistro(Usuario usuario);  
    public void eliminarRegistro(Usuario usuario);  
    public void actualizarRegistro(Usuario usuario);  
    public Usuario obtenerRegistro(Integer id);  
    public List<Usuario> obtenerRegistros();  
}
```

```
package strus.edu.mx.model;
```

```
import java.util.List;  
import strus.edu.mx.entity.Usuario;
```

```
public class UsuarioModelImpl implements IUserioModel{
```

```
    @Override  
    public void crearRegistro(Usuario usuario) {
```

```
    }
```

```
    @Override  
    public void eliminarRegistro(Usuario usuario) {
```

```
    }
```

```
    @Override  
    public void actualizarRegistro(Usuario usuario) {
```

```
    }
```

```
    @Override  
    public Usuario obtenerRegistro(Integer id) {
```

```
        return null;  
    }
```

```
    @Override  
    public List<Usuario> obtenerRegistros() {
```

```
        return null;  
    }
```

```
}
```

# Construcción del proyecto Struts 2 - Capa Model

```
package strus.edu.mx.model;

import java.util.List;
import org.hibernate.HibernateException;
import org.hibernate.Session;
import org.hibernate.SessionFactory;
import org.hibernate.cfg.Configuration;
import strus.edu.mx.entity.Usuario;

public class UsuarioModelImpl implements IUsuarioModel{
    private SessionFactory sf;
    private Session s;

    @Override
    public void crearRegistro(Usuario usuario) {
        try {
            sf = new Configuration().configure().buildSessionFactory();
            s = sf.openSession();
            s.beginTransaction();
            s.save(usuario);
            s.getTransaction().commit();
            s.close();
            sf.close();
        } catch (HibernateException e) {
            System.out.println("Error: " + e.getMessage());
        }
    }

    @Override
    public void eliminarRegistro(Usuario usuario) {
        try {
            sf = new Configuration().configure().buildSessionFactory();
            s = sf.openSession();
            s.beginTransaction();
            s.delete(usuario);
            s.getTransaction().commit();
            s.close();
            sf.close();
        } catch (HibernateException e) {
            System.out.println("Error: " + e.getMessage());
        }
    }
}
```

```
@Override
public void actualizarRegistro(Usuario usuario) {
    try {
        sf = new Configuration().configure().buildSessionFactory();
        s = sf.openSession();
        s.beginTransaction();
        s.update(usuario);
        s.getTransaction().commit();
        s.close();
        sf.close();
    } catch (HibernateException e) {
        System.out.println("Error: " + e.getMessage());
    }
}

@Override
public Usuario obtenerRegistro(Integer id) {
    try {
        sf = new Configuration().configure().buildSessionFactory();
        s = sf.openSession();
        Usuario usuario = (Usuario) s.get(Usuario.class, id);
        s.close();
        sf.close();
        return usuario;
    } catch (HibernateException e) {
        System.out.println("Error al recuperar los registros: " + e.getMessage());
    }
    return null;
}

@Override
public List<Usuario> obtenerRegistros() {
    try {
        sf = new Configuration().configure().buildSessionFactory();
        s = sf.openSession();
        List<Usuario> lista = s.createCriteria(Usuario.class).list();
        s.close();
        sf.close();
        return lista;
    } catch (HibernateException e) {
        System.out.println("Error al recuperar los registros: " + e.getMessage());
    }
    return null;
}
```

}

# Construcción del proyecto Struts 2 - Capa Service

```
package strus.edu.mx.service.base;

import java.util.List;

public interface IRegistroServiceBase<T> {
    public void crearRegistro(T registro);
    public void eliminarRegistro(T registro);
    public void actualizarRegistro(T registro);
    public T obtenerRegistro(Integer id);
    public List<T> obtenerRegistros();
}
```

Se crea una interfaz genérica para manipular cualquier tipo de objeto, y se implementan las cinco operaciones básicas del CRUD.

En caso de requerir más métodos que compartan las mismas características con los objetos, se declaran los métodos en esta interfaz.

# Construcción del proyecto Struts 2 - Capa Service

```
package strus.edu.mx.service;

import strus.edu.mx.entity.Usuario;
import strus.edu.mx.service.base.IRegistroServiceBase;

public interface IUsuarioService extends IRegistroServiceBase<Usuario>{
}
```

Se extiende de la interfaz genérica y se especifica el objeto.

```
package strus.edu.mx.service;

import java.util.List;
import strus.edu.mx.entity.Usuario;
import strus.edu.mx.model.IUsuarioModel;
import strus.edu.mx.model.UsuarioModelImpl;

public class UsuarioServiceImpl implements IUsuarioService{
    private IUsuarioModel model = new UsuarioModelImpl();

    @Override
    public void crearRegistro(Usuario registro) {
        model.crearRegistro(registro);
    }

    @Override
    public void eliminarRegistro(Usuario registro) {
        model.eliminarRegistro(registro);
    }

    @Override
    public void actualizarRegistro(Usuario registro) {
        model.actualizarRegistro(registro);
    }

    @Override
    public Usuario obtenerRegistro(Integer id) {
        return model.obtenerRegistro(id);
    }

    @Override
    public List<Usuario> obtenerRegistros() {
        return model.obtenerRegistros();
    }
}
```

Se realiza la instancia al modelo

# Construcción del proyecto Struts 2 - Capa Controller

```
package strus.edu.mx.controller.base;
import com.opensymphony.xwork2.ActionSupport;
import java.util.ArrayList;
import java.util.List;
import strus.edu.mx.entity.base.IRegistroEntityBase;
import strus.edu.mx.service.base.IRegistroServiceBase;

public class RegistroControllerBase<T extends IRegistroEntityBase> extends ActionSupport {
    private T registro;
    private List<T> listaRegistro;
    private IRegistroServiceBase<T> service;

    protected void resetItemLista(){
        listaRegistro = new ArrayList<>();
        listaRegistro = service.obtenerRegistros();
    }

    public String eliminarRegistro(){
        service.eliminarRegistro(registro);
        resetItemLista();
        return "success";
    }

    public String crearRegistro(){
        if(registro.getId() > 0){
            service.actualizarRegistro(registro);
        }else{
            service.crearRegistro(registro);
        }

        resetItemLista();
        return "success";
    }

    public T getRegistro() {
        return registro;
    }

    public void setRegistro(T registro) {
        this.registro = registro;
    }

    public void setService(IRegistroServiceBase<T> service) {
        this.service = service;
        resetItemLista();
    }

    public List<T> getListaRegistro() {
        return listaRegistro;
    }

    public void setListaRegistro(List<T> listaRegistro) {
        this.listaRegistro = listaRegistro;
    }
}
```

Se crea una clase abstracta genérica para manipular cualquier tipo de objeto, y se implementa un objeto de tipo genérico y el servicio de tipo genérico.

Al trabajar con diferentes tipos de objetos, se realiza el extends a esta clase para hacer uso de los métodos y objeto genérico.

La clase ActionSupport es un extends que se realiza a partir de Struts 2. Y todos los métodos deben ser de tipo público y String.

# Construcción del proyecto Struts 2 - Capa Controller

```
package strus.edu.mx.controller;

import strus.edu.mx.controller.base.RegistroControllerBase;
import strus.edu.mx.entity.Usuario;
import strus.edu.mx.service.IUsuarioService;
import strus.edu.mx.service.UsuarioServiceImpl;

public class UsuarioController extends RegistroControllerBase<Usuario> {

    private IUsuarioService service;
    private Usuario usuario;

    public UsuarioController() {
        service = new UsuarioServiceImpl();
        super.setService(service);
        super.resetItemList();
    }

    public Usuario getUsuario() {
        return usuario;
    }

    public void setUsuario(Usuario usuario) {
        this.usuario = usuario;
        super.setRegistro(usuario);
    }
}
```

Se crea el servicio de tipo IUsuarioService y se setea en la clase padre dentro del constructor.

```
public UsuarioController() {
    service = new UsuarioServiceImpl();
    super.setService(service);
    super.resetItemList();
}
```

Se genera el set y get de Usuario para que Struts 2 pueda hacer el seteo de los datos desde los archivos jsp que invoque el UsuarioController.

Se extiende de la clase genérica y se especifica el objeto.

# Construcción del proyecto Struts 2 - Archivo struts.xml

The image displays the project structure of 'appStruts2' on the left and the 'struts.xml' configuration file on the right. Red arrows indicate the mapping between XML configurations and files: 'listarUsuario.jsp' (line 11), 'crearUsuario.jsp' (line 16), and 'editarUsuario.jsp' (line 33). A blue arrow points from 'RegistroControllerBase.java' (line 20) to the 'class' attribute in the 'crearUsuarioAction' configuration. A yellow box on the right explains that the 'UsuarioController' class is declared and inherits methods from the 'RegistroControllerBase' class.

**Project Structure (Left):**

- appStruts2
  - Web Pages
    - META-INF
    - WEB-INF
    - pages
      - content
        - crearUsuario.jsp
        - editarUsuario.jsp
        - listarUsuario.jsp
        - index.jsp
  - Source Packages
    - strus.edu.mx.controller
      - UsuarioController.java
    - strus.edu.mx.controller.base
      - RegistroControllerBase.java
    - strus.edu.mx.entity.base
    - strus.edu.mx.model
    - strus.edu.mx.service
    - strus.edu.mx.service.base
    - strus.edu.mx.utilerias
  - Test Packages
    - <default package>
  - Other Sources
    - src/main/resources
      - <default package>
        - hibernate.cfg.xml
        - struts.xml
    - META-INF
      - persistence.xml
  - Generated Sources (annotations)
  - Dependencies
  - Java Dependencies
  - Project Files

**struts.xml Content (Right):**

```
1 <!DOCTYPE struts PUBLIC
2 "-//Apache Software Foundation//DTD Struts Configuration 2.0//EN"
3 "http://struts.apache.org/dtds/struts-2.0.dtd">
4
5 <!-- Author: informatica -->
6
7 <struts>
8
9   <package name="default" namespace="/" extends="struts-default">
10     <action name="listarUsuarioAction"
11       class="strus.edu.mx.controller.UsuarioController">
12       <result>pages/content/listarUsuario.jsp</result>
13     </action>
14
15     <action name="crearUsuarioActionForm">
16       <result>pages/content/crearUsuario.jsp</result>
17     </action>
18
19     <action name="crearUsuarioAction"
20       class="strus.edu.mx.controller.UsuarioController"
21       method="crearRegistro">
22       <result>pages/content/listarUsuario.jsp</result>
23     </action>
24
25     <action name="eliminarUsuarioAction"
26       class="strus.edu.mx.controller.UsuarioController"
27       method="eliminarRegistro">
28       <result>pages/content/listarUsuario.jsp</result>
29     </action>
30
31     <action name="editarUsuarioActionForm"
32       class="strus.edu.mx.controller.UsuarioController">
33       <result>pages/content/editarUsuario.jsp</result>
34     </action>
35
36     <action name="editarUsuarioAction"
37       class="strus.edu.mx.controller.UsuarioController"
38       method="crearRegistro">
39       <result>pages/content/listarUsuario.jsp</result>
40     </action>
41   </package>
42 </struts>
```

**Código del archivo struts.xml**

Se declara la clase UsuarioController y por herencia, se ocupan los métodos del controller base.

# Código del archivo struts.xml

```
<!DOCTYPE struts PUBLIC
"-//Apache Software Foundation//DTD Struts Configuration 2.0//EN"
"http://struts.apache.org/dtds/struts-2.0.dtd">
```

```
<!-- Author: informatica -->
```

```
<struts>
```

```
    <package name="default" namespace="/" extends="struts-default">
        <action name="listarUsuarioAction"
            class="strus.edu.mx.controller.UsuarioController">
            <result>pages/content/listarUsuario.jsp</result>
        </action>
```

```
        <action name="crearUsuarioActionForm">
            <result>pages/content/crearUsuario.jsp</result>
        </action>
```

```
        <action name="crearUsuarioAction"
            class="strus.edu.mx.controller.UsuarioController"
            method="crearRegistro">
            <result>pages/content/listarUsuario.jsp</result>
        </action>
```

```
        <action name="eliminarUsuarioAction"
            class="strus.edu.mx.controller.UsuarioController"
            method="eliminarRegistro">
            <result>pages/content/listarUsuario.jsp</result>
        </action>
```

```
        <action name="editarUsuarioActionForm"
            class="strus.edu.mx.controller.UsuarioController">
            <result>pages/content/editarUsuario.jsp</result>
        </action>
```

```
        <action name="editarUsuarioAction"
            class="strus.edu.mx.controller.UsuarioController"
            method="crearRegistro">
            <result>pages/content/listarUsuario.jsp</result>
        </action>
    </package>
```

```
</struts>
```



# index.jsp

```
<%@taglib prefix="s" uri="/struts-tags" %>
<%@page contentType="text/html" pageEncoding="UTF-8"%>
<!DOCTYPE html>
<html>

    <head>
        <meta charset=UTF-8">
        <title>Gestión de Usuario</title>
        <link rel="stylesheet" href="${pageContext.request.contextPath}/webjars/bootstrap/5.2.2/css/bootstrap.min.css">
        <link rel="stylesheet" href="${pageContext.request.contextPath}/webjars/bootstrap-icons/1.10.2/font/bootstrap-icons.css">

        <script type="text/javascript" src="/resources/js/jquery-3.3.1.min.js"></script>
        <script type="text/javascript" src="${pageContext.request.contextPath}/webjars/bootstrap/5.2.2/js/bootstrap.min.js"></script>
    </head>
    <body>
        <div class="page-header">
            <nav class="navbar navbar-expand-sm bg-dark navbar-dark">
                <ul class="navbar-nav">
                    <li class="nav-item active">
                        <a class="navbar-brand" href="#"><i class="alarm"></i>Gestión de Usuario</a>
                    </li>
                    <li class="nav-item">
                        <s:a href="/listarUsuarioAction">
                            <span class="nav-link"><i class="bi-card-list"></i>Listar</span>
                        </s:a>
                    </li>
                    <li class="nav-item">
                        <s:a href="/crearUsuarioActionForm">
                            <span class="nav-link bi-clipboard-data">Crear</span>
                        </s:a>
                    </li>
                    <li class="nav-item">
                        <s:a href="#">
                            <span class="nav-link">Municipio</span>
                        </s:a>
                    </li>
                    <li class="nav-item">
                        <s:a href="#">
                            <span class="nav-link">Localidad</span>
                        </s:a>
                    </li>
                </ul>
            </nav>
        </div>
    </body>
</html>
```

# listarUsuario.jsp

```
<%@taglib prefix="s" uri="/struts-tags"%>
<%@page contentType="text/html" pageEncoding="UTF-8"%>
<!DOCTYPE html>
<html>

    <head>
        <meta charset=UTF-8>
        <title>Listar Usuario</title>
        <link rel="stylesheet"
href="${pageContext.request.contextPath}/webjars/bootstrap/5.2.2/css/bootstrap.min.css">
        <link rel="stylesheet"
href="${pageContext.request.contextPath}/webjars/bootstrap-icons/1.10.2/font/bootstrap-icons.css">
        <script type="text/javascript" src="/resources/js/jquery-3.3.1.min.js"></script>
        <script type="text/javascript"
src="${pageContext.request.contextPath}/webjars/bootstrap/5.2.2/js/bootstrap.min.js"></script>
    </head>
    <body>
        <div class="page-header">
            <nav class="navbar navbar-expand-sm bg-dark navbar-dark">
                <ul class="navbar-nav">
                    <li class="nav-item active">
                        <s:a class="navbar-brand" href="actionPrincipal">
                            <span class="navbar-brand">Gestión de Usuario</span>
                        </s:a>
                    </li>

                    <li class="nav-item">
                        <s:a href="actionCrearUsuarioAction">
                            <span class="nav-link"> Crear</span>
                        </s:a>
                    </li>
                </ul>
            </nav>
        </div>
```

```
<div class="container">
    <div class="row">
        <!-- LIST -->
        <div class="col-md-12">
            <table class="table table-bordered table-hover">
                <thead>
                    <tr>
                        <th class="bi-circle-fill">Id</th>
                        <th class="bi-file-play">Nombre</th>
                        <th>Contraseña</th>
                        <th>Fecha Creación</th>
                        <th>Fecha Actualización</th>
                        <th>Fecha Eliminación</th>
                        <th>Opciones</th>
                    </tr>
                </thead>
                <tbody>
                    <s:iterator value="listaRegistro" var="registro" status="estado">
                        <tr>
                            <td><s:property value="idUsuario"></s:property></td>
                            <td><s:property value="nombreUsuario"></s:property></td>
                            <td><s:property value="contraseña"></s:property></td>
                            <td><s:property value="fechaCreación"></s:property></td>
                            <td><s:property value="fechaActualización"></s:property></td>
                            <td><s:property value="fechaEliminación"></s:property></td>
                            <td>
                                <s:a action="editarUsuarioActionForm">
                                    <s:param name="usuario.idUsuario" value="idUsuario"></s:param>
                                    <s:param name="usuario.nombreUsuario" value="nombreUsuario"></s:param>
                                    <span class="btn bg-info">Editar</span>
                                </s:a>

                                <s:a action="eliminarUsuarioAction">
                                    <s:param name="usuario.idUsuario" value="idUsuario"></s:param>
                                    <span class="btn bg-danger">Eliminar</span>
                                </s:a>
                            </td>
                        </tr>
                    </s:iterator>
                </tbody>
            </table>
        </div>
    </div>
</div>
```

# crearUsuario.jsp

```

<%@taglib prefix="s" uri="/struts-tags"%>
<%@page contentType="text/html" pageEncoding="UTF-8"%>
<!DOCTYPE html>
<html>

    <head>
        <meta charset=UTF-8>
        <title>Crear Usuario</title>
        <link rel="stylesheet"
href="${pageContext.request.contextPath}/webjars/bootstrap/5.2.2/css/bootstrap.min.cs
s">

        <script type="text/javascript" src="./resources/js/jquery-3.3.1.min.js"></script>
        <script type="text/javascript"
src="${pageContext.request.contextPath}/webjars/bootstrap/5.2.2/js/bootstrap.min.js"></
script>

    </head>

    <body>
        <div class="page-header">
            <nav class="navbar navbar-expand-sm bg-dark navbar-dark">
                <ul class="navbar-nav">
                    <li class="nav-item active">
                        <s:a class="navbar-brand" href="actionPrincipal">
                            <span class="navbar-brand">Gestión de Usuario</span>
                        </s:a>
                    </li>
                    <li class="nav-item">
                        <s:a href="/listarUsuarioAction">
                            <span class="nav-link">Listar Usuario</span>
                        </s:a>
                    </li>
                </ul>
            </nav>
        </div>
    </body>
</html>

```

```
<div class="container">
  <div class="row">

    <!-- FORM-->
    <div class="col-md-12">
      <div class="card">
        <div class="card-body">
          <s:form action="crearUsuarioAction" method="POST">

            <div class="form-group">
              <s:textfield name="usuario.nombreUsuario"
                size="200px"
                placeholder="Ingrese el nombre"
                cssClass="form-control">

            </s:textfield>
          </div>

          <div class="form-group">
            <s:textfield name="usuario.contraseña"
              size="200px"
              placeholder="Ingrese la contraseña"
              cssClass="form-control">

            </s:textfield>
          </div>

          <s:submit type="button" value="Crear Registro"
            cssClass="btn btn-success btn-block" />
        </s:form>
      </div>
    </div>
  </div>
</div>
</body>
</html>
```

# editarUsuario.jsp

```
<%@taglib prefix="s" uri="/struts-tags"%>
<%@page contentType="text/html" pageEncoding="UTF-8"%>
<!DOCTYPE html>
<html>
    <head>
        <meta charset="UTF-8">
        <title>Editar Usuario</title>
        <link rel="stylesheet"
href="${pageContext.request.contextPath}/webjars/bootstrap/5.2.2/css/bootstrap.min.css">
        <script type="text/javascript" src="/resources/js/jquery-3.3.1.min.js"></script>
        <script type="text/javascript"
src="${pageContext.request.contextPath}/webjars/bootstrap/5.2.2/js/bootstrap.min.js"></script>
    </head>
    <body>
        <div class="page-header">
            <nav class="navbar navbar-expand-sm bg-dark navbar-dark">
                <ul class="navbar-nav">
                    <li class="nav-item active">
                        <s:a class="navbar-brand" href="actionPrincipal">
                            <span class="navbar-brand">Gestión de Usuario</span>
                        </s:a>
                    </li>
                    <li class="nav-item">
                        <s:a href="actionCrearUsuarioAction" >
                            <span class="nav-link"> Crear</span>
                        </s:a>
                    </li>
                    <li class="nav-item">
                        <s:a href="listarUsuarioAction" >
                            <span class="nav-link">Mostrar</span>
                        </s:a>
                    </li>
                </ul>
            </nav>
        </div>
```

```
<div class="container">
    <div class="row">
        <!-- FORM-->
        <div class="col-md-12">
            <div class="card">
                <div class="card-body">
                    <s:form action="editarUsuarioAction" method="POST">
                        <div class="form-group" >
                            <s:textfield name="usuario.idUsuario"
                                size="200px"
                                cssClass="form-control">
                        </s:textfield>
                    </div>
                    <div class="form-group">
                        <s:textfield name="usuario.nombreUsuario"
                            size="200px"
                            cssClass="form-control">
                    </s:textfield>
                </div>
                <div>
                    <s:submit value="Actualizar Registro"
                        cssClass="btn btn-success btn-block" />
                </div>
            </s:form>
        </div>
    </div>
</div>
</body>
</html>
```

# Archivo hibernate.cfg.xml

```
<hibernate-configuration>
  <session-factory>
    <property name="hibernate.dialect">org.hibernate.dialect.MySQL8Dialect</property>
    <property name="hibernate.connection.driver_class">com.mysql.cj.jdbc.Driver</property>
    <property name="hibernate.connection.url">jdbc:mysql://localhost/banco</property>
    <property name="hibernate.connection.username">root</property>
    <property name="hibernate.connection.password">secreto</property>
    <property name="hibernate.connection.pool_size">1</property>
    <property name="hibernate.show_sql">true</property>
    <property name="hibernate.current_session_context_class">thread</property>
    <property name="hibernate.hbm2ddl.auto">update</property>
    <mapping class="strus.edu.mx.entity.Usuario"></mapping>
  </session-factory>
</hibernate-configuration>
```

# Dependencias Maven

```
<dependency>
  <groupId>org.apache.struts</groupId>
  <artifactId>struts2-core</artifactId>
  <version>2.1.8</version>
</dependency>

<!-- https://mvnrepository.com/artifact/org.hibernate/hibernate-core -->
<dependency>
  <groupId>org.hibernate</groupId>
  <artifactId>hibernate-core</artifactId>
  <version>5.6.12.Final</version>
  <type>jar</type>
</dependency>

<!-- https://mvnrepository.com/artifact/mysql/mysql-connector-java -->
<dependency>
  <groupId>mysql</groupId>
  <artifactId>mysql-connector-java</artifactId>
  <version>8.0.31</version>
</dependency>
```

```
<!-- https://mvnrepository.com/artifact/org.webjars/bootstrap -->
<dependency>
  <groupId>org.webjars</groupId>
  <artifactId>bootstrap</artifactId>
  <version>5.2.2</version>
</dependency>

<!-- https://mvnrepository.com/artifact/org.webjars.npm/bootstrap-icons -->
<dependency>
  <groupId>org.webjars.npm</groupId>
  <artifactId>bootstrap-icons</artifactId>
  <version>1.10.2</version>
</dependency>

<!-- https://mvnrepository.com/artifact/org.webjars.bower/jquery -->
<dependency>
  <groupId>org.webjars.bower</groupId>
  <artifactId>jquery</artifactId>
  <version>3.6.1</version>
</dependency>
```

# web.xml

```
<?xml version="1.0" encoding="UTF-8"?>
<web-app version="3.1" xmlns="http://xmlns.jcp.org/xml/ns/javaee"
xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
xsi:schemaLocation="http://xmlns.jcp.org/xml/ns/javaee
http://xmlns.jcp.org/xml/ns/javaee/web-app_3_1.xsd">
    <filter>
        <filter-name>struts2</filter-name>
        <filter-class>org.apache.struts2.dispatcher.ng.filter.StrutsPrepareAndExecuteFilter</filter-class>
    </filter>
    <filter-mapping>
        <filter-name>struts2</filter-name>
        <url-pattern>/*</url-pattern>
    </filter-mapping>

    <session-config>
        <session-timeout>
            30
        </session-timeout>
    </session-config>
    <welcome-file-list>
        <welcome-file>pages/index.jsp</welcome-file>
    </welcome-file-list>
</web-app>
```