

## Documentación del Proyecto

### Descripción General

Este proyecto es una aplicación web desarrollada en Java que utiliza Servlets y AJAX para manejar peticiones del cliente. Su propósito es procesar archivos CSV, realizar validaciones y gestionar datos de manera eficiente.

### Tecnologías Utilizadas

- **Java 8:** Lenguaje principal del proyecto.
- **Servlets:** Para manejar las peticiones HTTP y la lógica del backend.
- **AJAX:** Para realizar solicitudes asíncronas desde el frontend.
- **HTML, CSS, JavaScript:** Para la interfaz de usuario.
- **Maven:** Gestión de dependencias y construcción del proyecto.
- **Base de datos:** Se utiliza una base de datos relacional (por definir en el script de BD).

### Estructura del Proyecto

```
src/  
├── main/  
│   ├── java/  
│   │   ├── com/previred/ejercicio/  
│   │   │   ├── dao/          # Acceso a datos (JDBC)  
│   │   │   │   ├── EmpleadoDAO.java  
│   │   │   ├── model/       # Modelos de datos  
│   │   │   │   ├── Empleado.java  
│   │   │   ├── servlets/    # Servlets para los endpoints  
│   │   │   │   ├── EmpleadoServlet.java  
│   │   │   │   ├── NominaServlet.java  
│   │   │   ├── util/        # Utilidades (BD, JSON, etc.)  
│   │   │   │   ├── DatabaseUtil.java  
│   │   ├── webapp/  
│   │   │   ├── WEB-INF/  
│   │   │   │   ├── web.xml    # Configuración de Servlets  
│   │   │   │   ├── index.html # Interfaz principal  
│   │   │   │   ├── js/  
│   │   │   │   │   ├── app.js    # Lógica AJAX/JS  
│   │   │   │   │   └── css/  
│   │   │   │   │       ├── styles.css    # Estilos básicos  
│   │   ├── test/            # Tests (opcional)  
└── resources/  
    └── application.properties
```

## Componentes Clave

El proyecto ejercicioPrevired está organizado en varios paquetes que separan las responsabilidades del código. A continuación, se describen las clases principales de cada paquete:

### Paquete: **com.previred.ejercicio.dao**

- EmpleadoDAO.java: Esta clase se encarga del acceso a los datos relacionados con los empleados. Implementa operaciones Crear, Eliminar, obtener todos los empleados, obtener por id, verificar el rut, utilizando JDBC para interactuar con la base de datos.

### Paquete: **com.previred.ejercicio.model**

- Empleado.java: Define el modelo de datos para un empleado, incluyendo atributos String nombre, String apellido, String rut, String cargo, BigDecimal salarioBase, BigDecimal bonos, BigDecimal descuentos, BigDecimal salarioFinal.
- . También incluye métodos getter y setter para cada atributo.

### Paquete: **com.previred.ejercicio.servlet**

- EmpleadoServlet.java: Maneja las solicitudes HTTP relacionadas con los empleados, como la creación y eliminación. Utiliza EmpleadoDAO para interactuar con la base de datos y devuelve respuestas en formato JSON.
- NominaServlet.java: Gestiona las solicitudes relacionadas con la nómina de empleados, como la generación de informes de salarios. También interactúa con EmpleadoDAO y devuelve datos en formato JSON.

### Paquete: **com.previred.ejercicio.util**

- DatabaseUtil.java: Proporciona métodos utilitarios para la Inicialización de tablas ,conexión y gestión de la base de datos, como la obtención de conexiones y el manejo de excepciones relacionadas con la base de datos.

### Archivo: src/main/webapp/index.html

Este archivo constituye la interfaz principal de la aplicación web. Incluye formularios para la gestión de empleados y la visualización de la nómina, y utiliza AJAX para comunicarse de manera asíncrona con los servlets del backend.

Esta estructura modular facilita el mantenimiento y la escalabilidad del proyecto, permitiendo una separación clara de responsabilidades entre la lógica de negocio, el acceso a datos y la presentación.

### **Instalación y Ejecución**

1. Clonar el repositorio.
  2. Importar el proyecto en un IDE compatible con Maven (IntelliJ, Eclipse, etc.).
  3. Configurar la conexión a la base de datos en web.xml o en un archivo de configuración.
  4. Compilar y ejecutar el proyecto en un servidor compatible con Servlets (Tomcat, Jetty, etc.).
  5. Acceder a <http://localhost:8080/ejercicioPrevired> en el navegador.
-