

PRÁCTICA 3 EXTRA UT2.2. Creación de componentes visuales

¡Lee atentamente la rúbrica al final de este documento!

Estás en una prueba de código en una entrevista de trabajo para una empresa puntera en el Desarrollo Software, te motiva trabajar en esta empresa porque el sueldo es más que jugoso y se ofrece un horario flexible y teletrabajo.

Te han pedido en la empresa que desarrolles una **interfaz de gestión de los empleados**. En la captura de requisitos funcionales habéis determinado que el sistema debe tener las siguientes funcionalidades:

- **Ventana Principal:** Muestra un menú con 3 botones para agregar un nuevo empleado, ver la lista de empleados y salir. La clase debe llamarse `VentanaPrincipal.java`.
- **Ventana de Alta de Empleado:** En esta ventana, se podrán introducir los siguientes datos: nombre, apellidos, departamento ("Ventas", "IT", "Recursos Humanos", usando un `JComboBox`) y si el empleado está activo o no (usando un `JCheckBox` personalizado). La clase debe llamarse `VentanaAgregarEmpleado`.
- **Ventana de Lista de Empleados:** Muestra la lista de empleados registrados en una tabla o en un área de texto. La clase debe llamarse `VentanaListaEmpleados`.

Debes añadir los siguientes componentes y funcionalidades:

- **Botones** (`JButton`): Para navegar entre ventanas y confirmar acciones (agregar, salir, etc.).
- **Cajas de Texto** (`JTextField`): Para introducir datos como nombre y apellidos con validación de campos.
- **Selector de Departamento** (`JComboBox`): Para seleccionar el departamento de un empleado.

- **Componente personalizado** (JCheckBox): Se personalizará un JCheckBox para que represente si el empleado está "Activo" o "Inactivo" cambiando de color de rojo a verde cuando se seleccione.
- Agregar **validaciones** para los campos obligatorios (por ejemplo, nombre y apellidos no deben estar vacíos).
- Guardar la información de los empleados en una estructura de datos (como un **ArrayList**), que luego se visualice en la ventana de lista.
- Puedes añadir más campos como la **fecha** de contratación.

Esta práctica evalúa los siguientes Resultados de Aprendizaje y Criterios de Evaluación:

RA1: Genera interfaces gráficos de usuario mediante editores visuales utilizando las funcionalidades del editor y adaptando el código generado	15 %	UD 1
a) Se han analizado las herramientas y librerías disponibles para la generación de interfaces gráficos.	10%	
b) Se ha creado un interfaz gráfico utilizando las herramientas de un editor visual.	15%	
c) Se han utilizado las funciones del editor para ubicar los componentes del interfaz.	10%	
d) Se han modificado las propiedades de los componentes para adecuarlas a las necesidades de la aplicación.	15%	
e) Se ha analizado el código generado por el editor visual.	10%	
f) Se ha modificado el código generado por el editor visual.	10%	
g) Se han asociado a los eventos las acciones correspondientes.	10%	
h) Se ha desarrollado una aplicación que incluye el interfaz gráfico obtenido.	20%	
RA2: Genera interfaces naturales de usuario utilizando herramientas visuales	10 %	UD 2

a) Se han identificado las herramientas disponibles para el aprendizaje automático relacionadas con las interfaces de usuario.	40%	
b) Se ha creado una interfaz natural de usuario utilizando las herramientas disponibles.	40%	

Se utilizará la siguiente rúbrica:

Rúbrica de Evaluación para la Interfaz de Gestión de Empleados

Criterios	1 puntos	0,5 puntos	0,25 puntos	0 puntos
1. Funcionalidades Principales (Ventana Principal, Ventana de Alta de Empleado, Ventana de Lista de Empleados)	Implementa todas las ventanas con los nombres de clase requeridos y con todas las funcionalidades básicas	Cumple con las funcionalidades principales pero falta alguno de los detalles menores o alguna funcionalidad secundaria	Las tres ventanas están presentes, pero presentan errores en funcionalidades críticas o en la nomenclatura	No cumple con los requisitos de creación de ventanas ni funcionalidades requeridas
2. Navegación y Botones (Flujo de ventanas y uso de botones)	Navegación entre ventanas fluida e intuitiva; botones claramente identificados y sin errores de navegación	Navegación completa, con algunos problemas de fluidez o de claridad en el etiquetado de botones	Navegación con varios errores o botones que no están claramente etiquetados o son confusos	Navegación ineficaz o inexistente; los botones no cumplen su función adecuadamente

3. Validación de Datos (Campos de nombre y apellidos)	Validaciones de campos obligatorios implementadas y funcionales; previene entradas vacías correctamente	Validación de campos presente, pero con algún error menor que no afecta gravemente la funcionalidad	Validación de campos con errores frecuentes o no verifica todos los campos requeridos	Sin validación de campos o funcionalidad incompleta para los campos obligatorios
4. Selector de Departamento (JComboBox)	JComboBox implementado, con opciones correctas y funcionalidad sin errores	Implementado con alguna funcionalidad menor faltante, pero funcional en general	Selector de departamento implementado, pero con opciones incorrectas o funcionalidad parcial	JComboBox ausente o sin funcionalidad correcta
5. Componente Personalizado (JCheckBox)	Implementación de JCheckBox personalizado completa, incluyendo el cambio de color dinámico	JCheckBox implementado con cambio de color, pero con detalles menores faltantes	Implementado sin cambio de color o funcionalidad incompleta	No implementa el JCheckBox personalizado según los requisitos
6. Estructura de Datos para Guardado (Uso de ArrayList)	Estructura de datos implementada correctamente para el almacenamiento de datos y funcionamiento estable	Uso de ArrayList correcto, pero con algunos detalles que afectan la visualización o estabilidad	Almacenamiento en ArrayList, pero sin persistencia de datos o visualización incorrecta	No se usa ninguna estructura de datos o implementación incorrecta que impide el guardado

DESARROLLO DE INTERFACES

DAM2

7. Visualización de Datos (En ventana de lista de empleados)	Visualización clara y completa de los empleados almacenados, con tabla o área de texto funcional	Visualización adecuada, con detalles de presentación mejorables o errores menores	Visualización parcial o incompleta de la lista de empleados	No presenta la lista de empleados o la presentación no es funcional
8. Aspecto y Estilo de la Interfaz (Diseño y apariencia)	Interfaz atractiva, bien organizada, con estilo visual consistente	Interfaz funcional y organizada, pero con detalles de estilo que pueden mejorarse	Interfaz aceptable pero con problemas de organización o apariencia poco clara	Interfaz desorganizada, con apariencia deficiente o sin estilo
9. Código Limpio y Nomenclatura (Buenas prácticas)	Código claro y organizado, con nomenclatura apropiada y comentarios útiles	Código mayormente organizado, con uso adecuado de nombres y algunos comentarios	Código con problemas de organización o con nomenclatura inconsistente	Código desorganizado o sin seguir buenas prácticas de nomenclatura
10. Creatividad y Extensión de Funcionalidades (Ej. añadir campo de fecha de contratación)	Añade funcionalidades extra útiles (como fecha de contratación) de manera correcta y funcional	Funcionalidades extra añadidas, pero con algunos errores menores o poco efectivas	Funcionalidades extra añadidas sin lograr el efecto deseado	No añade ninguna funcionalidad adicional