

PARTE PRÁCTICA

Requerimientos previos:

- Java JDK16 instalado en el sistema
- IDE Eclipse, Netbeans o IntelliJ Idea configurado correctamente con JavaFX.
- SceneBuilder
- Servidor MySQL
- JasperSoft Studio

Contexto:

Queremos una desarrollar una aplicación en Java que sea capaz de gestionar las calificaciones de los alumnos de la clase de 2DAM que almacena los datos en una base de datos MySQL. Vamos a realizar el diseño básico que tendría el interfaz usando **JavaFX**.

Necesitamos poder tratar además del nombre y los apellidos de los alumnos, su correo, dni, telefono, fecha de nacimiento, localidad y cada una de las notas globales de cada módulo (AD, SGE, DI, PMDM, PSP, EIE y HLC), para así poder obtener de un vistazo todas las calificaciones.

Vamos a disponer de solo una vista en nuestra aplicación con los siguientes controles:

- Una tabla para mostrar a todos los alumnos de la clase.
- Un formulario formado por cajetines de texto y un botón que sirva para introducir un nuevo alumno con sus notas correspondientes en la tabla.
- Un botón “descargar notas” que permita generar un pdf con todas las notas para usarlo en el claustro.
- Un botón “salir” para salir.

La disposición de los elementos en la vista debe ser lo más adecuada posible teniendo en cuenta la usabilidad de la aplicación.

Se pide:

1) Crear la vista principal de la aplicación en JavaFX usando SceneBuilder, teniendo en cuenta que todos los controles han de disponer de ayuda contextual cuando el ratón se pasa sobre él, indicando que función realizan.

Añadir a vuestra respuesta lo siguiente:

1. el archivo .fxml creado,
2. Una captura en la que se vea la ventana

(2 puntos)

2) Crear la clase Alumno necesaria para mostrar los datos en la vista principal y crear el controlador de la aplicación de manera que al inicio de la ejecución añada al menos 8 alumnos de prueba en la tabla **(no es necesario que los cargue desde la base de datos)**, y permita añadir alumnos durante la ejecución usando el formulario creado.

Todos los campos serán obligatorios y el controlador deberá validar que las notas estén en el intervalo [0-10] y permita 1 decimal de precisión.

La funcionalidad de la tabla debe ser que, al pulsar sobre un alumno, aparezca un texto en un pequeño modal que muestre:

- Todos sus datos
- La nota media si tiene aprobados todos los módulos.
- El número de módulos suspensos (solo si tuviera algún módulo suspenso)

Hay que tener en cuenta que todo el código deberá estar debidamente documentado.

(2,5 puntos)

3) Crear una base de datos en tu servidor MySQL local que tenga dos tablas, una para los alumnos y otra para las calificaciones (relación 1 a muchos) y sirva para almacenar la información de los alumnos e introducir los mismos alumnos de prueba que en el apartado anterior con sus respectivas notas (1 punto)

Adjuntar capturas de pantalla con el contenido de la base de datos creada.

4) Crear un informe en Jasper Studio que sirva para generar un informe de las notas de todos los alumnos y crear con él un archivo pdf con todas las notas de los alumnos. Se valorará especialmente el diseño gráfico del informe.

Adjuntar ese archivo pdf creado directamente desde JasperSoft

(3 puntos)

5) Queremos modificar el informe y que además de mostrar todas las notas de los alumnos, aparezca una columna final con la nota media de cada alumno (la media de las notas de todos sus módulos)

(1,5 punto)

Subid vuestra solución a Github en un nuevo repositorio y adjuntar exclusivamente su enlace (nada de archivos zip, rar o 7z).

Añadid también varias capturas de pantalla donde se vea la salida del programa.

Además del correcto funcionamiento, se valorará en vuestra solución la documentación y calidad del código propuesto.

Si se detecta el más mínimo indicio de plagio, copia o uso de herramientas como CoPilot de forma indebida, los exámenes involucrados quedarán descalificados.

Se prohíbe el uso del chat de teams durante el transcurso del examen. Si se detecta su uso, se retirará el examen de los alumnos implicados.

Autoevaluación

Nota esperada del
ejercicio práctico: