Informe Técnico

Sistema de Evaluación Interactivo con Taxonomía Cognitiva en Java

**1. Introducción**

En el contexto actual de la educación, las herramientas tecnológicas juegan un papel fundamental en la evaluación del aprendizaje. El presente proyecto, desarrollado en lenguaje Java, tiene como propósito diseñar un sistema interactivo que permita aplicar pruebas personalizadas a estudiantes, considerando los niveles de la Taxonomía de Bloom. A su vez, los docentes pueden gestionar un banco de preguntas clasificadas según asignatura y nivel cognitivo.

El sistema se ejecuta mediante una interfaz gráfica basada en Swing, que facilita la navegación y usabilidad para usuarios sin conocimientos técnicos avanzados. Las preguntas se leen desde un archivo externo (preguntas.csv) y se presentan al usuario de manera dinámica, mostrando el resultado final con estadísticas relevantes.

Los niveles cognitivos considerados se alinean con los seis escalones clásicos de Bloom:

- Recordar

- Comprender

- Aplicar

- Analizar

- Evaluar

- Crear

Esto permite generar evaluaciones que van más allá de la memorización, fomentando habilidades de razonamiento y análisis. Al concluir una evaluación, se ofrece un resumen del puntaje obtenido, promoviendo la retroalimentación formativa.

**1.2 Objetivos del Informe**

- Documentar la arquitectura modular del sistema, organizada en componentes backend y frontend.

- Describir los principales elementos del sistema, especificando sus funciones y cómo colaboran para cumplir con los objetivos pedagógicos.

- Justificar las decisiones de diseño adoptadas, como el uso del paradigma orientado a objetos, el tratamiento de archivos CSV, y la separación de lógica visual y funcional.

- Reflexionar sobre el funcionamiento del sistema, proponiendo futuras mejoras como el almacenamiento de resultados, personalización de evaluaciones o integración con bases de datos.

**2. Descripción de la Solución**

**2.1 Arquitectura General**

El sistema ha sido diseñado en una estructura modular y orientada a objetos, compuesta por dos capas principales:

- Backend (backend): Encapsula los datos y la lógica del dominio, incluyendo la representación de preguntas y la lectura desde archivo.

- Frontend (frontend): Gestiona la interfaz de usuario mediante formularios Swing, con pantallas como bienvenida, selección de asignatura, selección de taxonomía, prueba y resultados.

Esta separación favorece la escalabilidad, mantenibilidad y futuras integraciones.

***Componentes principales:***

- PreguntaVF: Clase que representa una pregunta del tipo Verdadero/Falso. Contiene los atributos principales: asignatura, taxonomía, enunciado, respuesta correcta y ponderación.

- LectorPreguntas: Clase que se encarga de leer las preguntas desde un archivo .csv, procesando los campos correctamente incluso si contienen comillas o separadores complejos.

- TaxonomiaApp: Clase principal (*main*) que inicia la aplicación gráfica mediante *SwingUtilities.invokeLater().*

2.2 Detalle de Componentes y Funcionalidades

**A) Clase PreguntaVF**

- Responsabilidad: Representar una pregunta de evaluación.

***Atributos:***

- asignatura: Categoría o materia de la pregunta.

- taxonomia: Nivel cognitivo (como "Recordar" o "Evaluar").

- enunciado: Texto de la pregunta.

- correcta: Valor booleano que indica si la afirmación es verdadera.

- ponderacion: Puntaje que aporta si se responde correctamente.

***Métodos:***

- Getters para cada atributo (*getAsignatura(), isCorrecta(), etc.*).

**B) Clase LectorPreguntas**

- Responsabilidad: Leer y parsear preguntas desde un archivo .csv.

***Características:***

- Utiliza expresiones regulares para dividir correctamente cada línea en sus partes, incluso si el enunciado contiene comas.

- Convierte la respuesta ("V" o "F") en un valor booleano.

- Devuelve una lista de objetos PreguntaVF que alimentan la lógica de evaluación en la interfaz gráfica.

**C) Interfaz Gráfica (Swing)**

***Pantallas principales:***

- SplashScreenForm: Pantalla de bienvenida.

- Selección de asignatura y nivel: Permiten al usuario elegir sobre qué materia y nivel desea ser evaluado.

- Evaluación: Se presentan preguntas una por una con botones de opción para responder.

- Resultados: Muestra el puntaje total obtenido y los detalles por cada nivel de taxonomía.

***Comportamiento:***

- Carga dinámica de preguntas según filtros.

- Actualización del puntaje acumulado tras cada respuesta.

- Transiciones fluidas entre formularios.

**2.3 Decisiones de Diseño**

- Uso de Java Swing: Se eligió por ser parte del JDK estándar, permitiendo una interfaz amigable sin dependencias externas.

- Lectura desde CSV: Simplifica la edición y carga de preguntas por parte de los docentes, sin necesidad de herramientas adicionales.

- Separación en capas: Permite escalar el sistema fácilmente, por ejemplo, reemplazando el archivo CSV por una base de datos en futuras versiones.

- Orientación a objetos: Cada entidad del sistema (pregunta, lector, interfaz) está encapsulada en su propia clase, facilitando la lectura y prueba del código.

**3. Conclusión**

El sistema de evaluación basado en Java cumple satisfactoriamente los objetivos educativos y técnicos planteados. Sus principales logros incluyen:

- Evaluación interactiva y personalizada: La interfaz permite que los estudiantes seleccionen asignatura y nivel de dificultad, recibiendo retroalimentación inmediata al finalizar.

- Gestión externa de preguntas: Los docentes pueden preparar bancos de preguntas simplemente editando un archivo .csv, promoviendo una implementación práctica y flexible.

- Diseño mantenible y escalable: Gracias a su estructura modular, el sistema puede ser ampliado fácilmente, integrando nuevas funcionalidades como:

- Almacenamiento de resultados por usuario

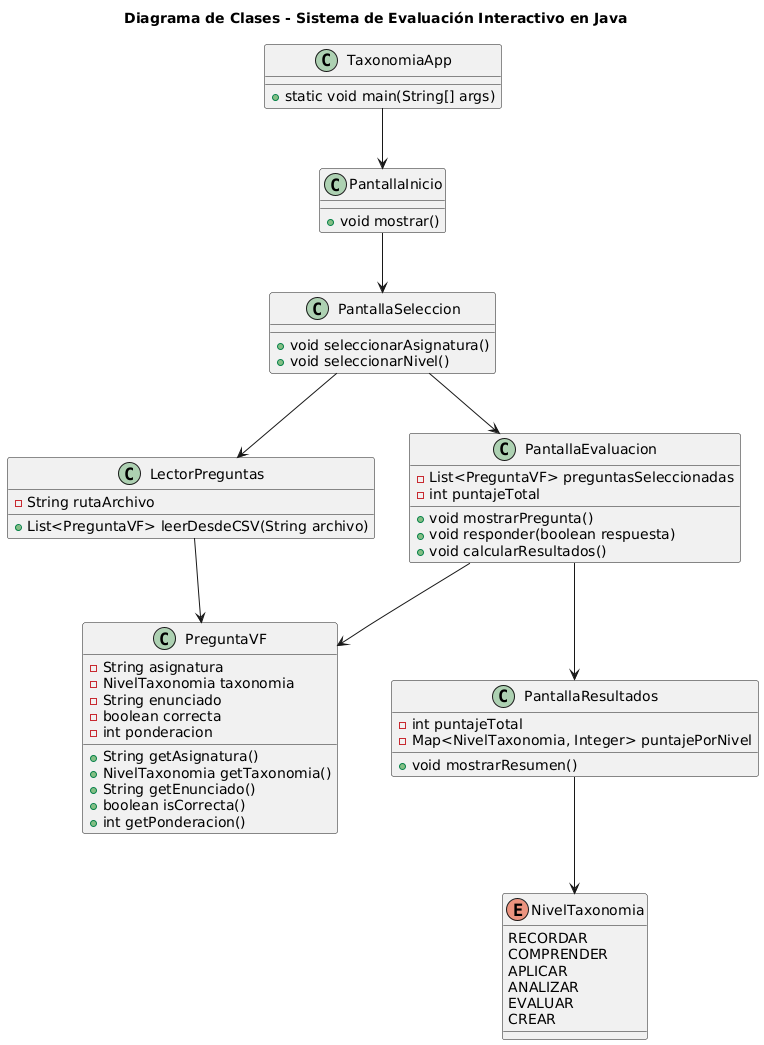
- Edición de preguntas desde la interfaz

- Soporte multilingüe

- Exportación de estadísticas a PDF o Excel

- Alineación con estándares pedagógicos: El uso de la Taxonomía de Bloom asegura que las evaluaciones midan no solo conocimientos, sino también competencias cognitivas superiores.

En suma, esta solución representa un paso concreto hacia la modernización de la evaluación educativa, con base en principios pedagógicos sólidos y tecnologías accesibles.

**Diagrama de Clases**