**VALIDACION DE ALGORITMOS DE SEGURIDAD**

INTEGRANTES: GRUPO 3

Pariente Merida Alejandro

Varela Quiroz Jorge Russell

Rojas Cossio Valeria Milka

Gonzales Vargas Isabel

Gutierrez Cadima Bricely Clariss

Tapia Leon Victor Hugo

**Introducción**

El desarrollo del sistema de procesamiento de datos implicó la selección de herramientas tecnológicas adecuadas para cada fase del proceso, desde la extracción y manipulación de datos hasta su almacenamiento y visualización. Este documento detalla cada una de las herramientas utilizadas, su propósito dentro del sistema y la justificación de su elección basada en criterios de eficiencia, compatibilidad y seguridad.

**Descripción del Sistema**

El sistema desarrollado cumple con los siguientes pasos principales:

1. **Extracción de Datos**: Se lee un archivo de texto (.txt) con información relevante que debe ser procesada.
2. **Parseo y Encriptación**: La información extraída se parsea para estructurar los datos y luego se encripta para garantizar su seguridad.
3. **Almacenamiento en Base de Datos**: Los datos procesados se guardan en una base de datos SQL Server.
4. **Visualización de Datos**: Se desarrolla una interfaz en WPF que permite consultar y visualizar los datos almacenados.

Cada una de estas etapas requiere el uso de herramientas específicas que optimicen el rendimiento y garanticen la seguridad del sistema.

**Herramientas Utilizadas y Justificación**

**1. SQL Server (Gestor de Base de Datos)**

**Uso:** Almacenamiento y gestión de los datos extraídos y procesados.

**Justificación:**

* Proporciona una solución robusta y escalable para la administración de datos.
* Ofrece seguridad en el almacenamiento y consultas eficientes mediante T-SQL.
* Compatible con herramientas de Microsoft, lo que facilita la integración con C# y WPF.
* Soporta transacciones y procedimientos almacenados, lo que optimiza el manejo de datos.
* Permite la implementación de medidas de seguridad avanzadas, como cifrado de datos y control de acceso a nivel de usuario.

**2. C# (Lenguaje de Programación para el Parser y la Encriptación)**

**Uso:** Desarrollo del programa que parsea los datos del archivo TXT y los encripta antes de almacenarlos en la base de datos.

**Justificación:**

* Lenguaje versátil y orientado a objetos, ideal para la manipulación de datos y algoritmos de encriptación.
* Buena integración con SQL Server y WPF, permitiendo un flujo de datos eficiente.
* Soporta librerías de encriptación como System.Security.Cryptography, garantizando la protección de la información.
* Permite la creación de programas eficientes con buen manejo de memoria y alto rendimiento en procesamiento de datos.
* Amplia comunidad y soporte, lo que facilita la resolución de problemas y la incorporación de nuevas funcionalidades.

**3. WPF (Windows Presentation Foundation) (Interfaz de Usuario)**

**Uso:** Creación de la interfaz gráfica donde se muestran los datos almacenados en la base de datos.

**Justificación:**

* Proporciona una experiencia visual atractiva y moderna para el usuario.
* Permite la separación del diseño y la lógica mediante XAML y el patrón MVVM.
* Integración nativa con C# y SQL Server, optimizando la visualización y manipulación de los datos.
* Soporte para gráficos avanzados y personalización de la interfaz mediante estilos y plantillas.
* Facilita la creación de interfaces responsivas y adaptables a diferentes tamaños de pantalla.

**Beneficios de la Elección Tecnológica**

La combinación de estas herramientas ofrece varios beneficios clave:

* **Seguridad**: Uso de encriptación en C# y medidas avanzadas en SQL Server para proteger los datos.
* **Eficiencia**: Procesamiento rápido de datos gracias a la optimización en C# y consultas eficientes en SQL Server.
* **Facilidad de Mantenimiento**: Separación clara de capas en la arquitectura del sistema, facilitando futuras actualizaciones.
* **Interoperabilidad**: Integración fluida entre SQL Server, C# y WPF, permitiendo un flujo de datos continuo y sin interrupciones.

**Conclusión**

La elección de estas herramientas se fundamenta en su compatibilidad y eficiencia para el desarrollo del sistema. SQL Server garantiza un almacenamiento seguro y eficiente, C# permite un procesamiento de datos flexible con capacidades de encriptación avanzadas, y WPF proporciona una interfaz visual moderna y adaptable. Gracias a esta combinación, el sistema desarrollado cumple con los objetivos de almacenamiento, procesamiento y visualización de los datos de manera efectiva, asegurando un rendimiento óptimo y una experiencia de usuario satisfactoria.

**Introducción**

El desarrollo del sistema de procesamiento de datos implicó la selección de herramientas tecnológicas adecuadas para cada fase del proceso, desde la extracción y manipulación de datos hasta su almacenamiento y visualización. Este documento detalla cada una de las herramientas utilizadas, su propósito dentro del sistema y la justificación de su elección basada en criterios de eficiencia, compatibilidad y seguridad.

**Descripción del Sistema**

El sistema desarrollado cumple con los siguientes pasos principales:

1. **Extracción de Datos**: Se lee un archivo de texto (.txt) con información relevante que debe ser procesada.
2. **Parseo y Encriptación**: La información extraída se parsea para estructurar los datos y luego se encripta para garantizar su seguridad.
3. **Almacenamiento en Base de Datos**: Los datos procesados se guardan en una base de datos SQL Server.
4. **Visualización de Datos**: Se desarrolla una interfaz en WPF que permite consultar y visualizar los datos almacenados.

Cada una de estas etapas requiere el uso de herramientas específicas que optimicen el rendimiento y garanticen la seguridad del sistema.

**Herramientas Utilizadas y Justificación**

**1. SQL Server (Gestor de Base de Datos)**

**Uso:** Almacenamiento y gestión de los datos extraídos y procesados.

**Justificación:**

* Proporciona una solución robusta y escalable para la administración de datos.
* Ofrece seguridad en el almacenamiento y consultas eficientes mediante T-SQL.
* Compatible con herramientas de Microsoft, lo que facilita la integración con C# y WPF.
* Soporta transacciones y procedimientos almacenados, lo que optimiza el manejo de datos.
* Permite la implementación de medidas de seguridad avanzadas, como cifrado de datos y control de acceso a nivel de usuario.

**2. C# (Lenguaje de Programación para el Parser y la Encriptación)**

**Uso:** Desarrollo del programa que parsea los datos del archivo TXT y los encripta antes de almacenarlos en la base de datos.

**Justificación:**

* Lenguaje versátil y orientado a objetos, ideal para la manipulación de datos y algoritmos de encriptación.
* Buena integración con SQL Server y WPF, permitiendo un flujo de datos eficiente.
* Permite la creación de programas eficientes con buen manejo de memoria y alto rendimiento en procesamiento de datos.
* Amplia comunidad y soporte, lo que facilita la resolución de problemas y la incorporación de nuevas funcionalidades.

**3. Algoritmo de Encriptación AES (Advanced Encryption Standard)**

**Uso:** Protección de los datos sensibles antes de ser almacenados en la base de datos.

**Justificación:**

* AES es un estándar de cifrado altamente seguro utilizado a nivel mundial.
* Ofrece cifrado simétrico, lo que lo hace eficiente en términos de velocidad y consumo de recursos.
* Compatible con la biblioteca System.Security.Cryptography de C#, facilitando su implementación.
* Protege la confidencialidad de los datos en tránsito y en reposo, asegurando que solo las partes autorizadas puedan acceder a ellos.

**4. WPF (Windows Presentation Foundation) (Interfaz de Usuario)**

**Uso:** Creación de la interfaz gráfica donde se muestran los datos almacenados en la base de datos.

**Justificación:**

* Proporciona una experiencia visual atractiva y moderna para el usuario.
* Permite la separación del diseño y la lógica mediante XAML y el patrón MVVM.
* Integración nativa con C# y SQL Server, optimizando la visualización y manipulación de los datos.
* Soporte para gráficos avanzados y personalización de la interfaz mediante estilos y plantillas.
* Facilita la creación de interfaces responsivas y adaptables a diferentes tamaños de pantalla.

**Beneficios de la Elección Tecnológica**

La combinación de estas herramientas ofrece varios beneficios clave:

* **Seguridad**: Uso de encriptación AES en C# y medidas avanzadas en SQL Server para proteger los datos.
* **Eficiencia**: Procesamiento rápido de datos gracias a la optimización en C# y consultas eficientes en SQL Server.
* **Facilidad de Mantenimiento**: Separación clara de capas en la arquitectura del sistema, facilitando futuras actualizaciones.
* **Interoperabilidad**: Integración fluida entre SQL Server, C# y WPF, permitiendo un flujo de datos continuo y sin interrupciones.

**Conclusión**

La elección de estas herramientas se fundamenta en su compatibilidad y eficiencia para el desarrollo del sistema. SQL Server garantiza un almacenamiento seguro y eficiente, C# permite un procesamiento de datos flexible con capacidades de encriptación avanzadas, y WPF proporciona una interfaz visual moderna y adaptable. Gracias a esta combinación, el sistema desarrollado cumple con los objetivos de almacenamiento, procesamiento y visualización de los datos de manera efectiva, asegurando un rendimiento óptimo y una experiencia de usuario satisfactoria.