Universidad Rafael Landívar

Facultad de Ingeniería

Ingeniería en Informática y sistemas

Manejo e implementación de archivos

Catedrático: Vivian Damaris Del Socorro Campos Gonzáles

***PROYECTO 1.***

Alison Gatica - 1136122

Camilo Solís - 231184

Guatemala, 16 de octubre del 2024

**Introducción**

Este proyecto tiene como objetivo desarrollar una herramienta de simulación de un sistema de mensajería, implementada en C#. La aplicación gestionará mensajes entre usuarios, almacenando todos los datos en archivos de texto en el directorio "C:/MEIA" pero no se permite el uso de estructuras de datos en memoria para almacenar información, solo para mostrarla en pantalla.

**Acceso al sistema y gestión de usuarios**

* El sistema verifica las credenciales al inicio y distinguirá mayúsculas de minúsculas. Si el usuario no existe, podrá crear uno nuevo, asignando roles de administrador para el primero y usuario para los demás.
* Los datos del usuario, como nombre, apellido, contraseña (cifrada), rol, fecha de nacimiento, teléfono y estatus, se almacenan en user.txt y se describen en desc\_user.
* Se requerirá un nivel mínimo de seguridad para las contraseñas, y el sistema evaluará la seguridad según los criterios almacenados en un archivo de configuración.

**Roles y permisos**

* Los usuarios con rol de administrador podrán agregar, buscar y modificar dar de baja a otros usuarios mientras que los usuarios regulares solo podrán cambiar su propia información o darse cuenta de baja.
* El estatus del usuario se controlará a través de un campo especifico y se reflejará en el sistema

La primera fase de este proyecto abarca la implementación del sistema de acceso, la gestión básica de usuarios y la funcionalidad de respaldo de información. El objetivo es proporcionar una base funcional del sistema de mensajería, enfocándose en el control de acceso, roles y estructura de archivos necesarios para gestionar los datos de manera efectiva.

**Interfaz**

**Login Admin info**

Interfaz de usuario gráfica, Aplicación

Descripción generada automáticamente Interfaz de usuario gráfica

Descripción generada automáticamente

**Admin Edit admin**

Interfaz de usuario gráfica

Descripción generada automáticamente Interfaz de usuario gráfica, Sitio web

Descripción generada automáticamente

**Edit user Search user**

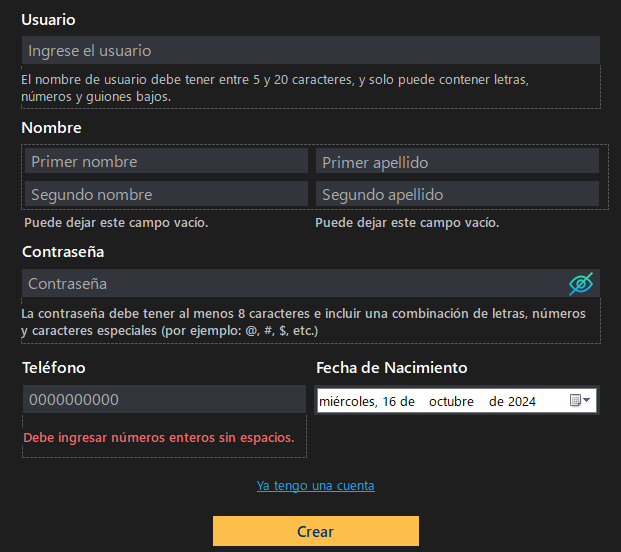
Interfaz de usuario gráfica, Sitio web

Descripción generada automáticamente Captura de pantalla de un celular

Descripción generada automáticamente

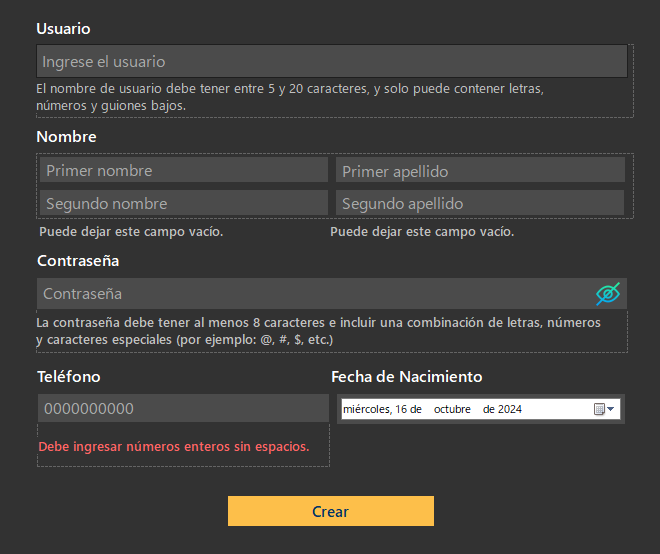
**User info Sign up**

Interfaz de usuario gráfica

Descripción generada automáticamente 

**User addUser**

Imagen que contiene Interfaz de usuario gráfica

Descripción generada automáticamente 

**Bibliotecas**

**Using MessageNest.Forms;y using MessageNest:**

Estas bibliotecas parecen ser específicas para tu proyecto, posiblemente personalizadas o de un paquete externo llamado MessageNest . Sin el contexto de MessageNest , podría deducirse que proporciona funcionalidades relacionadas con mensajería o comunicaciones y que contiene controles personalizados para formularios de Windows.

**using System:**

Es la biblioteca principal del sistema .NET, proporcionando tipos y funcionalidades básicas, como tipos de datos primitivos, gestión de excepciones, consola, entre otros.

**using System.CodeDom:**

Pertenece al espacio de nombres de System y permite generar y compilar código en tiempo de ejecución. Es útil para crear aplicaciones que generan código fuente dinámicamente, compilan o manejan clases de código de manera programática.

**using System.Collections.Generic:**

Proporciona clases de colección genérica, como List<T>, Dictionary<TKey, TValue>, entre otras. Es esencial para el manejo de colecciones de datos de manera eficiente y segura en cuanto a tipos.

**using System.ComponentModel:**

Proporciona clases que permiten interactuar con los componentes en tiempo de diseño y en tiempo de ejecución, como la implementación de eventos y propiedades enlazables. Es útil en aplicaciones que requieren funcionalidades de notificación de cambios, como formularios de Windows.

using System.Data:

Contiene clases para manejar datos y conectarse a bases de datos. Permite manipular tablas, filas y columnas en un contexto desconectado, como el uso de DataTable, DataSet, y DataView.

using System.Drawing:

Ofrece funcionalidades para manipular gráficos, imágenes y colores. Esta biblioteca es útil para crear y manipular elementos visuales, como dibujar formas, imágenes y texto en controles de Windows.

**using System.Linq:**

LINQ (Language Integrated Query) permite realizar consultas sobre colecciones y otros tipos de datos. Facilita la filtración, selección y proyección de datos de una manera muy concisa y legible.

**using System.Runtime.InteropServices:**

Proporciona una serie de clases que permiten la interoperabilidad con código no administrado, es decir, llamadas a funciones de bibliotecas externas o sistemas operativos, como las API de Windows.

**using System.Text:**

Incluye clases para trabajar con textos, especialmente para manipular codificaciones de caracteres, como UTF-8 o ASCII, y el uso de StringBuilderpara operaciones de concatenación de cadenas de manera eficiente.

**using System.Threading.Tasks:**

Permite trabajar con tareas asincrónicas y operaciones paralelas, facilitando el manejo de la concurrencia en aplicaciones, lo cual es útil para mejorar el rendimiento en procesos que pueden ejecutarse en paralelo.

**using System.Windows.Forms:**

Proporciona clases para crear aplicaciones de escritorio basadas en formularios de Windows, como botones, cuadros de texto y otros controles gráficos.

**using System.IO:**

Contiene clases para trabajar con archivos y flujos de datos, permitiendo la lectura y escritura de archivos, la creación de directorios y el manejo de archivos y directorios.

**Conclusiones**

* El proyecto demuestra como se puede administrar un sistema de mensajería sin depender de bases de datos complejas, aprovechando archivos de texto para el almacenamiento de datos.
* La asignación de roles diferenciados permite el control de acceso a las funciones del sistema de manera efectiva ya que los administradores tienen capacidades avanzadas como las gestión de usuarios y respaldos, mientras que los usuarios regulares se limitan a gestionar su propia información.
* Al depender exclusivamente de archivos de texto para el almacenamiento de datos, el sistema puede enfrentar problemas de rendimiento y manejo de datos en escenarios con grandes volúmenes de usuarios o mensajes.
* Este proyecto proporciona una solución de mensajería práctica, efectiva y segura, ideal para aplicaciones pequeñas y medias. La elección de almacenamiento de archivos, la gestión de roles y la validación de contraseñas contribuyen a un sistema robusto que es fácilmente adaptable y escalable.

**Recomendaciones**

* Si el sistema llega a manejar un gran numero de usuarios y mensajes estaría bien implementar técnicas para optimizar el acceso y almacenamiento de archivos como el uso de índices o almacenamiento en cache.
* Para proteger más el acceso al sistema, hay que considerar los controles de sesión, como expiraciones automáticas o un límite de intentos fallidos.
* Implementando ciertas recomendaciones el sistema de mensajería no solo será más seguro y fácil de usar, sino que también podrá gestionar datos de manera mas eficiente y estar mejor preparado.

**Manual de usuario**

<https://www.canva.com/design/DAE_C9w_3uQ/UHH_qbadI1TWQE0i93VZcg/edit?utm_content=DAE_C9w_3uQ&utm_campaign=designshare&utm_medium=link2&utm_source=sharebutton>

**Referencias**

[**https://learn.microsoft.com/es-es/dotnet/standard/class-library-overview**](https://learn.microsoft.com/es-es/dotnet/standard/class-library-overview)