TRABAJO FINAL PROGRAMACION DE SOFTWARE

Autores:

Jhonatan Toro Hurtado

Alejandro Castro Velásquez

Caso de estudio:

Los Torres Ferreteros SAS

Institución Universitaria ITM

Medellín

2024

**Requerimientos Funcionales**

* El sistema debe mostrar un formulario splash de 4 segundos al momento de iniciar la aplicación que contenga el título del proyecto, logo, desarrolladores y fecha de creación para brindar una introducción visual y profesional al sistema, lo que mejora la experiencia inicial del usuario y refuerza la identidad del proyecto.
* Debe tener un menú que permita navegar entre los diferentes formularios facilitando el acceso a las funcionalidades principales del sistema, mejorando la organización y experiencia del usuario al interactuar con las diferentes secciones.
* El sistema debe contar con un Inicio de sesión para asegura que solo usuarios autorizados puedan acceder al sistema, garantizando la seguridad e integridad de los datos administrados.
* La funcion de agregar usuarios es necesario para que el sistema pueda expandirse y para que el sistema pueda adaptarse al cambio de empleados que se pueda generar.
* Modificar usuarios permite mantener actualizados los datos de logeo y proporcionar una seguridad mas efectiva
* Eliminar usuarios facilita la gestión de permisos y datos, garantizando la limpieza de registros obsoletos o innecesarios.
* Consultar usuario por ID, esta consulta específica mejora la eficiencia en la búsqueda y recuperación de información, permitiendo un acceso rápido y directo a los datos del usuario.
* Agregar empleados pues esta funcionalidad es clave para gestionar el personal de la empresa, permitiendo mantener un registro de los empleados activos.
* Modificar empleados el poder atualizar los datos de los empleados asegura la confiabilidad y exactitud de la información almacenada en el sistema.
* Eliminar empleados, permitir la eliminación de empleados facilita la gestión de registros, especialmente cuando un empleado deja de pertenecer a la empresa.
* Listar los empleados ya que esto nos proporciona una vista consolidada y organizada de todos los registros disponibles, facilitando la supervisión y auditoría.
* Consultar empleado por documento, la búsqueda por documento permite localizar rápidamente un registro específico, optimizando tiempos y recursos.
* Agregar nuevos préstamos es fundamental para gestionar el flujo de implementos alquilados.
* Modificar préstamos debido a que permite corregir o actualizar los detalles de un préstamo en caso de que se detecten errores o cambios en las condiciones acordadas.
* La eliminación de préstamos ayuda a mantener un historial limpio y preciso, eliminando registros innecesarios o erróneos.
* Listar préstamos proporcionando una visión general del estado de todos los préstamos, facilitando el control y la toma de decisiones.
* Consultar préstamos por ID para poder acceder de manera directa a un registro específico, agilizando las consultas.
* Las vistas realizadas deben tener una opción de cancelar la acción que se está realizando, siempre y cuando esta no se haya finalizado todavía. Esto mejora la usabilidad del sistema al permitir que los usuarios reviertan acciones no deseadas antes de que se procesen definitivamente.

**Requerimientos No Funcionales**

* El sistema debe desarrollarse en C# utilizando ASP.NET como framework web para el backend pues ofrece un entorno robusto y escalable para el desarrollo de aplicaciones web, permitiendo aprovechar las características avanzadas del lenguaje C# para implementar la lógica del negocio.
* El sistema debe desarrollarse en HTML, CSS y JavaScript en un proyecto ASP.NET para el frontend para garantizar una interfaz moderna y compatible con la mayoría de los navegadores web, mejorando la experiencia del usuario.
* Como ORM se utilizará Entity Framework para la conexión y la realización de las transacciones en la base de datos, simplificando la interacción con esta, reduciendo el código necesario para las interacciones y asegurando un acceso más eficiente y seguro a los datos.
* El almacenamiento de datos debe realizarse en una base de datos relacional con SQL Server ya que es una solución confiable y ampliamente utilizada que garantiza la integridad, disponibilidad y seguridad de los datos en sistemas empresariales.
* La interfaz debe ser intuitiva y estar diseñada para facilitar el acceso rápido a las funciones del sistema mejorando la productividad y la satisfacción del usuario al minimizar la curva de aprendizaje y reducir errores operativos.
* La arquitectura del proyecto debe ser un modelo de programación por capas con el fin de separar las responsabilidades de cada componente del sistema, lo que facilita el mantenimiento, la escalabilidad y la reutilización del código.