

PONTIFICIA UNIVERSIDAD CATÓLICA DE
VALPARAÍSO FACULTAD DE INGENIERÍA ESCUELA
DE INGENIERÍA INFORMÁTICA

Proyecto FarmaSalud

Asignatura:

Estructura de datos.

Fecha de entrega:

11/06/2024

Profesor:

Dr. (c) Rafael Mellado S.

Integrantes:

Francisco Castro.

Joaquín Llanos.

Esteban Martínez.

Juan Pablo Pizarro.

Joaquín Saldivia.

Resumen:

El proyecto FarmaSalud, desarrollado por estudiantes de Ingeniería Informática de la Pontificia Universidad Católica de Valparaíso, consiste en una aplicación de gestión de inventarios para una cadena de farmacias a nivel nacional. Utilizando estructuras de datos en C, la aplicación permite el registro y control de productos, farmacias y proveedores, la gestión de stock en tiempo real, la generación de alertas de caducidad y desabastecimiento, y la elaboración de órdenes de compra. La solución mejora la eficiencia operativa y la precisión en la gestión de inventarios, asegurando un mejor servicio al cliente y la sostenibilidad del negocio.

Índice:

Resumen:	2
Índice:	3
Introducción:	4
Requerimiento de datos.	6
Planificación.....	23
Conclusiones	24
Anexo	26

Introducción:

La empresa FarmaSalud es una cadena de farmacias con presencia en todo el país. Esta empresa maneja un extenso catálogo de medicamentos y productos, cada uno identificado por un código único de 10 dígitos. La red de farmacias de FarmaSalud está distribuida en diversas ciudades y regiones, con cada establecimiento manteniendo su propio inventario y capacidad de almacenamiento, que varía según la ubicación y el tamaño de la tienda. En promedio, cada farmacia puede almacenar hasta 10,000 unidades de productos.

La gestión del inventario en FarmaSalud presenta varias interrogantes, tales como el control de lotes y fechas de caducidad, la distribución de productos, el seguimiento de los envíos de los proveedores, el abastecimiento, las alertas de este mismo y el análisis de datos para ver qué decisión tomar.

Para abordar estas problemáticas, se ha desarrollado un proyecto utilizando estructuras de datos en el lenguaje de programación C. Este mismo está diseñado para registrar productos con información detallada, gestionar múltiples farmacias y sus inventarios respectivos, controlar el stock en tiempo real, registrar ventas y actualizar el inventario, generar alertas de caducidad próxima y desabastecimiento, controlar productos que requieren receta médica, generar órdenes de compra a proveedores basadas en niveles de stock y demanda.

El desarrollo de este proyecto incluye la implementación de diversas estructuras de datos, como arreglos, listas enlazadas ya sean simples o dobles, árboles binarios de búsqueda, así como algoritmos de ordenamiento y búsqueda que facilitan el análisis de datos.

En este informe se presenta una descripción detallada del proceso de desarrollo de la aplicación, las metodologías utilizadas, las funcionalidades implementadas, y los resultados obtenidos. Se muestran también las dificultades durante el desarrollo y las soluciones adoptadas para superarlas, además se enfatizan en los supuestos para el desarrollo del programa.

Análisis

En esta sección se describen las estructuras de datos y los requerimientos de los datos que se utilizarán en la aplicación.

Usamos lista simple porque proporcionan flexibilidad en el tamaño, permitiendo que los datos crezcan o disminuyan dinámicamente sin necesidad de un tamaño fijo. Además, las operaciones de inserción y eliminación son eficientes, especialmente en el principio de la lista, y las listas simples aprovechan la memoria de manera dinámica, ocupando espacio solo para los elementos presentes. Su implementación y mantenimiento son relativamente sencillos, facilitando la gestión de datos de manera eficiente.

Usamos lista doble porque permite el recorrido de manera bidireccional (para atrás y para adelante), lo que facilita el recorrido. Esto mejora los procesos tanto en la eliminación como al agregar nodos, ya que no es necesario hacer todo el recorrido para modificarla.

Usamos un arreglo en los productos porque los arreglos dan acceso directo y eficiente a través de los índices, lo que permite realizar búsquedas y actualizaciones de manera rápida. Esta eficiencia es importante en aplicaciones de inventario donde se manejan grandes cantidades de productos y datos. También la implementación de un arreglo facilita el entendimiento del Código, haciendo que el mantenimiento del código a futuro sea más ameno. Además, los arreglos permiten una mejor gestión de la memoria, ya que se puede controlar el tamaño de este mismo. tamaño [10000] fijo.

Usamos un árbol de búsqueda porque los árboles permiten realizar búsquedas, inserciones y eliminaciones de manera eficiente, al mantener los proveedores ordenados, se simplifica la generación de reportes y la implementación de funciones para que actúen sobre este. Los árboles también proporcionan flexibilidad en las consultas, permitiendo búsquedas rápidas con diferentes criterios y campos. Esto optimiza la gestión y el mantenimiento de la información de proveedores en el sistema.

Supuestos para la correcta ejecución del código:

- a) Se asume que el usuario ingresa correctamente los datos en el menú.
- b) Se asume que cada producto contempla un único proveedor asociado a sí mismo y que éste siempre tiene stock para reponer.
- c) Se asume que las funciones de modificación son utilizadas en casos de mal tipeo al crear; por lo tanto, en ventas, podría cambiarse el código de producto, el nombre, fecha, etc. pero dentro del sistema, si un producto se cobró de más, simplemente se rehará la boleta y se agrega nuevamente el producto al stock manualmente. Posteriormente se elimina la boleta errónea.
- d) Se asume que los envíos llegarán en menos de un año.

Requerimiento de datos.

La estructura principal del sistema es FarmaSalud, que contiene dos listas: una lista doblemente enlazada de farmacias y una lista simplemente enlazada de proveedores. Esta estructura centraliza la gestión de todas las farmacias y proveedores dentro del sistema.

El Nodo Farmacia representa un nodo en una lista doblemente enlazada de farmacias. Cada nodo contiene un puntero a una estructura Farmacia y punteros a los nodos anterior y siguiente. Esta estructura permite recorrer las farmacias tanto hacia adelante como hacia atrás en la lista.

La estructura Farmacia almacena información específica de cada farmacia, incluyendo su ID, códigos de ciudad y región, capacidad de almacenamiento y nombre. Además, contiene un puntero a su inventario de productos, una lista de órdenes de compra realizadas a proveedores y una lista de ventas realizadas.

El arreglo Productos maneja un inventario de productos mediante un arreglo de punteros a productos y un índice de la primera posición libre. Esto facilita la organización y acceso a los productos dentro de una farmacia.

La estructura Producto define los atributos de un producto, incluyendo su código, precio, número de entrega, indicador de si requiere receta, stock disponible, umbral de stock, nombre, descripción, categoría, fecha de caducidad, nombre del lote, un puntero al proveedor y un puntero al arreglo de productos.

El Nodo Proveedores representa un nodo en una lista simplemente enlazada de proveedores. Cada nodo contiene un puntero a una estructura Proveedor y un puntero al siguiente nodo en la lista.

La estructura Proveedor almacena información sobre un proveedor, incluyendo su número telefónico, nombre, correo electrónico y un puntero a un árbol binario de lotes. Esta estructura organiza la información de cada proveedor y sus lotes de productos.

El Nodo Lotes representa un nodo en un árbol binario de lotes. Cada nodo contiene un puntero a un lote de productos y punteros a los nodos izquierdo y derecho del árbol.

La estructura Lote define los atributos de un lote de productos, incluyendo un puntero al producto dentro del lote, código del lote, precio, cantidad de productos en el lote, nombre del lote y fecha de caducidad del lote.

El NodoVenta representa un nodo en una lista simplemente enlazada de ventas. Cada nodo contiene un puntero al siguiente nodo en la lista y un puntero a una estructura Venta.

La estructura Venta almacena información sobre una venta realizada, incluyendo el código del producto vendido, cantidad vendida, número de boleta, monto pagado, indicador si el cliente trajo receta, nombre del producto vendido, RUT del cliente y fecha de la venta.

El Producto Orden define un producto dentro de una orden de compra, incluyendo su código, cantidad y un puntero al siguiente producto en la orden.

La estructura Orden Compra almacena la información de una orden de compra, incluyendo su identificador, monto comprado, fecha de llegada y una lista de productos en la orden.

El Nodo Orden Compra representa un nodo en una lista simplemente enlazada de órdenes de compra. Cada nodo contiene un puntero a una estructura OrdenCompra y un puntero al siguiente nodo en la lista.

cación son utilizadas en casos de mal tipeo al crear; por lo tanto, en ventas, podría cambiarse el código de producto, el nombre, fecha, etc., pero dentro de

Diseño:

En esta sección se detalla la solución en términos de las estructuras de datos y las funciones necesarias para manejar dichos datos.

1) mostrarProductos.

Recorre el inventario de productos de una farmacia seleccionada y muestra los detalles de cada producto. Los detalles incluyen el código del producto, nombre, descripción, categoría, fecha de caducidad (día, mes, año), precio, si requiere receta (indicando "Sí" o "No"), stock disponible y umbral. Asegura que toda la información relevante sobre cada producto sea claramente presentada al usuario, facilitando la gestión y revisión del inventario.

2) mostrarFarmacias.

Recorre la lista de farmacias registradas y muestra los detalles de cada una. Los detalles incluyen el ID de la farmacia, nombre, código de ciudad, código de región y capacidad de almacenamiento. Presenta de manera clara toda la información relevante sobre cada farmacia, facilitando su identificación y gestión dentro del sistema.

3) mostrarProveedores.

Recorre la lista de proveedores registrados y muestra los detalles de cada uno. Los detalles incluyen el ID del proveedor, nombre, email y número telefónico. Proporciona una visión clara y detallada de los proveedores disponibles, facilitando la revisión y gestión de sus datos de contacto.

4) mostrarLotes.

Recorre el árbol de lotes de productos y muestra los detalles de cada lote. Los detalles incluyen el código del lote, nombre, fecha de caducidad (día, mes, año), precio y cantidad de productos en el lote. Asegura que toda la información relevante sobre cada lote sea claramente presentada, facilitando su gestión y control.

5) mostrarVentas.

Recorre la lista de ventas registradas en una farmacia seleccionada y muestra los detalles de cada venta. Los detalles incluyen el nombre del producto vendido, la cantidad vendida, el número de boleta, el monto pagado, el RUT del cliente, si el cliente trajo una receta (indicando "Sí" o "No") y la fecha de venta (día, mes y año). Asegura que toda la información relevante sobre cada transacción sea claramente presentada al usuario, facilitando la revisión de las ventas realizadas.

6) mostrarProductosBajoUmbral.

Recorre el inventario de productos y muestra aquellos cuyo stock está por debajo del 50% del umbral definido. Los detalles mostrados incluyen el código del producto, nombre, descripción, categoría, fecha de caducidad (día, mes, año), precio, stock disponible y umbral. Ayuda a identificar los productos que necesitan ser reabastecidos, facilitando la gestión del inventario.

7) mostrarOrdenesCompra.

Recorre la lista de órdenes de compra y muestra los detalles de cada orden relacionada con una farmacia seleccionada. Los detalles incluyen el ID de la orden, monto comprado, fecha de llegada (día, mes, año) y los productos incluidos en la orden. Asegura que toda la información relevante sobre las órdenes de compra sea claramente presentada, facilitando el seguimiento y control de reabastecimientos.

8) mostrarProductosVendidosConReceta.

Recorre la lista de ventas y muestra los productos vendidos que requerían receta en una farmacia seleccionada. Los detalles incluyen el nombre del producto vendido, cantidad vendida, número de boleta, monto pagado, RUT del cliente y la fecha de venta (día, mes, año). Facilita la revisión de productos que requieren un control más estricto debido a la necesidad de receta médica.

9) mostrarProductosCaducanFecha.

Recorre el inventario de productos y muestra aquellos que caducan en una fecha específica. Los detalles incluyen el código del producto, nombre, descripción, categoría, fecha de caducidad (día, mes, año), precio, stock disponible y umbral. Ayuda a identificar productos próximos a caducar, permitiendo tomar decisiones sobre su venta o eliminación.

10) mostrarMenu.

Muestra un menú interactivo con todas las opciones disponibles para gestionar farmacias, productos, proveedores, lotes, ventas y órdenes de compra. El menú presenta las opciones de manera clara y estructurada, permitiendo al usuario acceder fácilmente a las diversas funcionalidades del sistema.

11) crearProducto.

Inicializa un nuevo producto con los detalles proporcionados, como código, precio, nombre, descripción, categoría, fecha de caducidad (día, mes, año), proveedor, nombre del lote, número de entrega, si requiere receta (indicando "Sí" o "No"), stock y umbral. Asegura que el producto sea creado con toda la información necesaria, listo para ser agregado al inventario de una farmacia.

12) crearArregloProductos.

Crea e inicializa un arreglo de productos vacío, listo para almacenar productos. Proporciona una estructura de almacenamiento inicial para gestionar el inventario de productos de una farmacia, permitiendo la adición de productos conforme se vayan creando.

13) crearFarmacia.

Inicializa una nueva farmacia con detalles como ID, código de ciudad, código de región, nombre, capacidad de almacenamiento, proveedores e inventario de productos. Asegura que la farmacia sea creada con toda la información necesaria, lista para ser integrada al sistema de gestión de farmacias.

14) crearLote.

Inicializa un nuevo lote con detalles como código del lote, producto asociado, precio, cantidad de productos, nombre del lote y fecha de caducidad (día, mes, año). Facilita la gestión de productos en lotes, asegurando que los productos sean correctamente organizados y controlados.

15) crearProveedor.

Inicializa un nuevo proveedor con detalles como ID, nombre, número telefónico y email. Asegura que el proveedor sea creado con toda la información necesaria, listo para ser agregado al sistema y relacionado con productos y farmacias.

16) crearVenta.

Inicializa una nueva venta con detalles como ID de farmacia, código del producto vendido, nombre del producto, cantidad vendida, número de boleta, monto pagado, RUT del cliente, si trajo receta (indicando "Sí" o "No"), y fecha de la venta (día, mes, año). Asegura que la venta sea registrada con toda la información necesaria, lista para ser agregada al historial de ventas de la farmacia.

17) crearOrdenCompra.

Inicializa una nueva orden de compra con detalles como ID de la orden, monto comprado, fecha de llegada (día, mes, año) y ID de la farmacia. Asegura que la orden de compra sea creada con toda la información necesaria, lista para ser procesada y gestionada en el sistema.

18) crearProductoOrden.

Inicializa un nuevo producto ordenado con detalles como código del producto y cantidad. Asegura que el producto ordenado sea creado con toda la información necesaria, listo para ser agregado a una orden de compra.

19) insertarFarmacia.

Inserta una nueva farmacia en la lista de farmacias, manteniendo la estructura de la lista doblemente enlazada. Asegura que la nueva farmacia sea correctamente integrada al sistema, facilitando su gestión y acceso.

20) insertarLote.

Inserta un nuevo lote en el árbol de lotes, asegurando la correcta organización basada en el código del lote. Permite la inclusión de nuevos lotes de productos en el sistema, facilitando su gestión y control.

21) insertarProveedor.

Inserta un nuevo proveedor en la lista de proveedores. Asegura que el nuevo proveedor sea correctamente agregado al sistema, manteniendo la organización de la lista de proveedores y facilitando su acceso y gestión.

22) insertarVenta.

Inserta una nueva venta en la lista de ventas de una farmacia, actualizando el inventario de productos. Asegura que la nueva venta sea registrada y que el stock del producto vendido sea decrementado adecuadamente, manteniendo el inventario actualizado.

23) insertarVentaEnFarmacia.

Inserta una nueva venta en la lista de ventas de una farmacia específica. Asegura que la venta sea correctamente asociada con la farmacia correspondiente, facilitando su gestión y revisión.

24) insertarOrdenCompra.

Inserta una nueva orden de compra en la lista de órdenes de compra. Asegura que la orden de compra sea correctamente integrada al sistema, facilitando su seguimiento y control de reabastecimientos.

25) buscar Producto.

Busca un producto en el inventario de una farmacia por su código y retorna el índice si lo encuentra, o -1 si no lo encuentra. Facilita la verificación de la existencia de productos en el inventario y su acceso rápido.

26) buscarFarmacia.

Busca una farmacia en la lista de farmacias por su ID y retorna un puntero a la farmacia si la encuentra, o NULL si no la encuentra. Facilita la identificación y selección de farmacias específicas dentro del sistema.

27) buscarProveedor.

Busca un proveedor en la lista de proveedores por su nombre y retorna un puntero al proveedor si lo encuentra, o NULL si no lo encuentra. Permite encontrar proveedores por su nombre, facilitando su gestión y relación con productos.

28) buscarProveedorPorId.

Busca un proveedor en la lista de proveedores por su ID y retorna un puntero al proveedor si lo encuentra, o NULL si no lo encuentra. Facilita la identificación de proveedores específicos por su ID.

29) buscarLote.

Busca un lote en el árbol de lotes por su código y retorna un puntero al lote si lo encuentra, o NULL si no lo encuentra. Permite localizar lotes específicos, útil para la gestión de caducidades y stock.

30) buscarVentaPorCódigo.

Busca una venta en la lista de ventas por el código del producto vendido y retorna un puntero a la venta si lo encuentra, o NULL si no lo encuentra. Facilita el seguimiento de ventas de productos específicos.

31) buscarVentaPorRut.

Busca una venta en la lista de ventas por el RUT del cliente y retorna un puntero a la venta si lo encuentra, o NULL si no lo encuentra. Permite encontrar ventas realizadas a un cliente específico, facilitando la revisión de su historial de compras.

32) buscarVentaPorBoleta.

Busca una venta en la lista de ventas por el número de boleta y retorna un puntero a la venta si lo encuentra, o NULL si no lo encuentra. Facilita la verificación de detalles de ventas específicas mediante el número de boleta.

33) buscarProductoMásVendido.

Recorre la lista de ventas y retorna un puntero a la venta con el producto más vendido. Facilita la identificación de los productos más populares o de mayor rotación en una farmacia.

34) eliminar Producto.

Elimina un producto del inventario de una farmacia, ajustando el arreglo de productos. Permite la remoción de productos obsoletos o no deseados del inventario, asegurando que el inventario se mantenga actualizado.

35) eliminar Farmacia.

Elimina una farmacia de la lista de farmacias, liberando la memoria utilizada por la farmacia y sus productos. Facilita la gestión y depuración del sistema al permitir la eliminación de farmacias que ya no son necesarias.

36) eliminarProveedor.

Elimina un proveedor de la lista de proveedores por su nombre. Permite la remoción de proveedores que ya no están en uso, asegurando que la lista de proveedores se mantenga actualizada.

37) eliminarProveedorPorId.

Elimina un proveedor de la lista de proveedores por su ID. Facilita la eliminación de proveedores específicos mediante su ID, manteniendo la organización de la lista de proveedores.

38) eliminarLote.

Elimina un lote del árbol de lotes por su código. Permite la remoción de lotes específicos, útil para la gestión de productos caducados y el mantenimiento del inventario.

39) eliminarVenta.

Elimina una venta de la lista de ventas por su número de boleta. Facilita la corrección y actualización de registros de ventas, asegurando que los datos de ventas se mantengan precisos.

40) eliminarProductosCaducados.

Recorre el inventario de productos y elimina aquellos que han caducado en una fecha específica. Ayuda a mantener el inventario actualizado y libre de productos caducados, asegurando la disponibilidad de productos frescos y seguros.

41) modificarProducto.

Modifica los detalles de un producto existente en el inventario de una farmacia. Los detalles modificables incluyen precio, stock, umbral, nombre, descripción, categoría, fecha de caducidad (día, mes, año), y si requiere receta (indicando "Sí" o "No"). Facilita

la actualización de información de productos, asegurando que los datos del inventario sean precisos y actuales.

42) modificarFarmacia.

Modifica los detalles de una farmacia existente, como capacidad de almacenamiento, códigos de ciudad y región, y nombre. Permite actualizar la información de farmacias según sea necesario, asegurando que los datos del sistema reflejen la situación actual de cada farmacia.

43) modificarLote.

Modifica los detalles de un lote existente en el árbol de lotes. Los detalles modificables incluyen precio, cantidad, nombre del lote y fecha de caducidad (día, mes, año). Facilita la actualización de información de lotes, asegurando que los datos reflejen con precisión la situación actual de cada lote.

44) modificarProveedor.

Modifica los detalles de un proveedor existente, como nombre, número telefónico y email. Permite la actualización de información de proveedores, asegurando que los datos de contacto y otros detalles sean precisos y actuales.

45) modificarVenta.

Modifica los detalles de una venta existente, como código de producto vendido, nombre del producto, cantidad vendida, número de boleta, monto pagado, RUT del cliente, si trajo receta (indicando "Sí" o "No"), y fecha de la venta (día, mes, año). Facilita la corrección y actualización de registros de ventas, asegurando que los datos sean precisos y reflejen la realidad de las transacciones.

46) modificarOrdenCompra.

Modifica los detalles de una orden de compra existente, incluyendo productos ordenados y fechas de llegada (día, mes, año). Permite la actualización de órdenes de compra para reflejar cambios en las necesidades de la farmacia, asegurando que los datos del sistema sean precisos y actuales.

47) reducirStocksProducto.

Reduce el stock de un producto en el inventario de una farmacia en una cantidad especificada. Asegura que el stock de productos sea decrementado adecuadamente tras una venta, manteniendo el inventario actualizado.

48) verificarReabastecimiento.

Verifica si un producto necesita reabastecimiento basándose en su promedio de ventas y su umbral. Ayuda a mantener un inventario adecuado evitando desabastecimientos, asegurando que los productos se repongan a tiempo.

49) calcularPromedioVentas.

Calcula el promedio de ventas de un producto en un mes específico. Proporciona información sobre la demanda de productos, facilitando decisiones informadas sobre inventario y reabastecimiento.

50) informeTendenciasVentas.

Genera un informe de tendencias de ventas para una farmacia en un rango de meses. Facilita el análisis de patrones de ventas y la toma de decisiones estratégicas basadas en las tendencias observadas.

51) productosAltaBajaRotación.

Identifica y muestra productos de alta y baja rotación en el inventario. Facilita la gestión del inventario, optimizando la reposición de productos según su rotación.

52) calcularMesesRestantes.

Calcula los meses restantes entre dos fechas. Es utilizada en la gestión de órdenes de compra para estimar tiempos de llegada y planificar reabastecimientos.

53) seguimientoEnvíos.

Proporciona seguimiento de envíos basándose en la fecha actual y la fecha estimada de llegada de una orden de compra. Facilita el control de pedidos y la planificación de reabastecimientos, asegurando que las órdenes lleguen a tiempo.

54) patronesEstacionales.

Analiza y muestra ventas en diferentes meses de un año específico para identificar patrones estacionales. Es útil para la planificación de inventario según temporadas de alta demanda, optimizando la disponibilidad de productos.

55) contarProductosDiferentes.

Cuenta el número de productos diferentes en el inventario. Proporciona una visión general de la variedad de productos en stock, facilitando la gestión del inventario.

56) obtenerCantidadDeProducto.

Retorna la cantidad total de un producto específico en el inventario. Facilita la gestión de stock al conocer la disponibilidad exacta de productos.

57) sumaCantidadDeProductosTotales.

Suma la cantidad de todos los productos en el inventario. Proporciona una visión del total de productos disponibles en la farmacia, facilitando la gestión del inventario.

58) imprimirDetallesVenta.

Imprime los detalles de una venta específica. Los detalles incluyen el código del producto vendido, cantidad vendida, número de boleta, monto pagado, nombre del producto, RUT del cliente y fecha de la venta (día, mes, año). Es útil para revisar la información detallada de una venta, asegurando que todos los datos relevantes sean claramente presentados.

59) calcularDemanda.

Calcula la demanda total de un producto específico en una farmacia recorriendo la lista de ventas y sumando la cantidad vendida de dicho producto.

60) calcularStockOptimo.

Calcula el stock óptimo de un producto en una farmacia específica basándose en la demanda total y el stock total del producto en todas las farmacias.

main.

La función principal del programa inicializa una farmacia, algunos proveedores, productos y ventas, estableciendo una estructura de datos inicial para el sistema. Luego, entra en un bucle que muestra un menú de opciones al usuario, permitiéndole realizar diversas acciones de gestión del inventario, farmacias, proveedores, lotes, ventas y órdenes de compra. El bucle se repite hasta que el usuario decide salir del programa. Esta función es el punto de entrada del programa y coordina la ejecución de las demás funciones según las opciones seleccionadas por el usuario.

Inicialización del Programa

Al iniciar el programa, se inicializan varias estructuras de datos principales. La estructura FarmaSalud centraliza la gestión de todas las farmacias y proveedores dentro del sistema. Cada farmacia tiene un inventario de productos representado por un arreglo de punteros a productos (arregloProductos). Los proveedores se representan mediante una estructura que incluye su información de contacto y un árbol binario de lotes (NodoLotes). Además, se establecen nodos para farmacias (NodoFarmacia) y ventas (NodoVenta), así como estructuras detalladas para productos, lotes y órdenes de compra.

Menú Principal

El programa presenta un menú interactivo que permite al usuario acceder a diferentes funcionalidades del sistema. Las opciones disponibles en el menú incluyen la capacidad de registrar nuevos productos, gestionar farmacias, controlar el stock en tiempo real, registrar ventas y actualizar automáticamente el inventario. El menú facilita la navegación por las diferentes funciones del programa, proporcionando una interfaz clara y estructurada para el usuario.

Funcionalidades de Registro y Gestión

El usuario puede registrar nuevos productos con detalles específicos como código, nombre, descripción, categoría, precio, proveedor, lote y fecha de caducidad. Además, se pueden agregar nuevas farmacias al sistema, cada una con su inventario y capacidad de almacenamiento específica. La gestión del stock en tiempo real permite al usuario ver y actualizar el stock de productos, asegurando que los niveles de inventario se mantengan precisos y actualizados.

Control de Lotes y Fechas de Caducidad

El programa incluye funcionalidades específicas para el control de lotes y fechas de caducidad. Permite registrar lotes de productos y llevar un seguimiento detallado de sus fechas de caducidad. Las alertas de caducidad próxima se generan automáticamente, ayudando a prevenir la venta de productos caducados. Esta función es esencial para mantener la seguridad y calidad de los productos ofrecidos a los clientes.

Generación de Órdenes de Compra

Para garantizar un suministro constante de productos, el programa permite generar órdenes de compra a proveedores basadas en niveles de stock y demanda. Estas órdenes se registran y se realiza el seguimiento de los envíos, asegurando que los productos lleguen a tiempo y en la cantidad necesaria. La función de órdenes de compra facilita la gestión de reabastecimientos y mejora la relación con los proveedores.

Registro de Ventas y Análisis de Datos

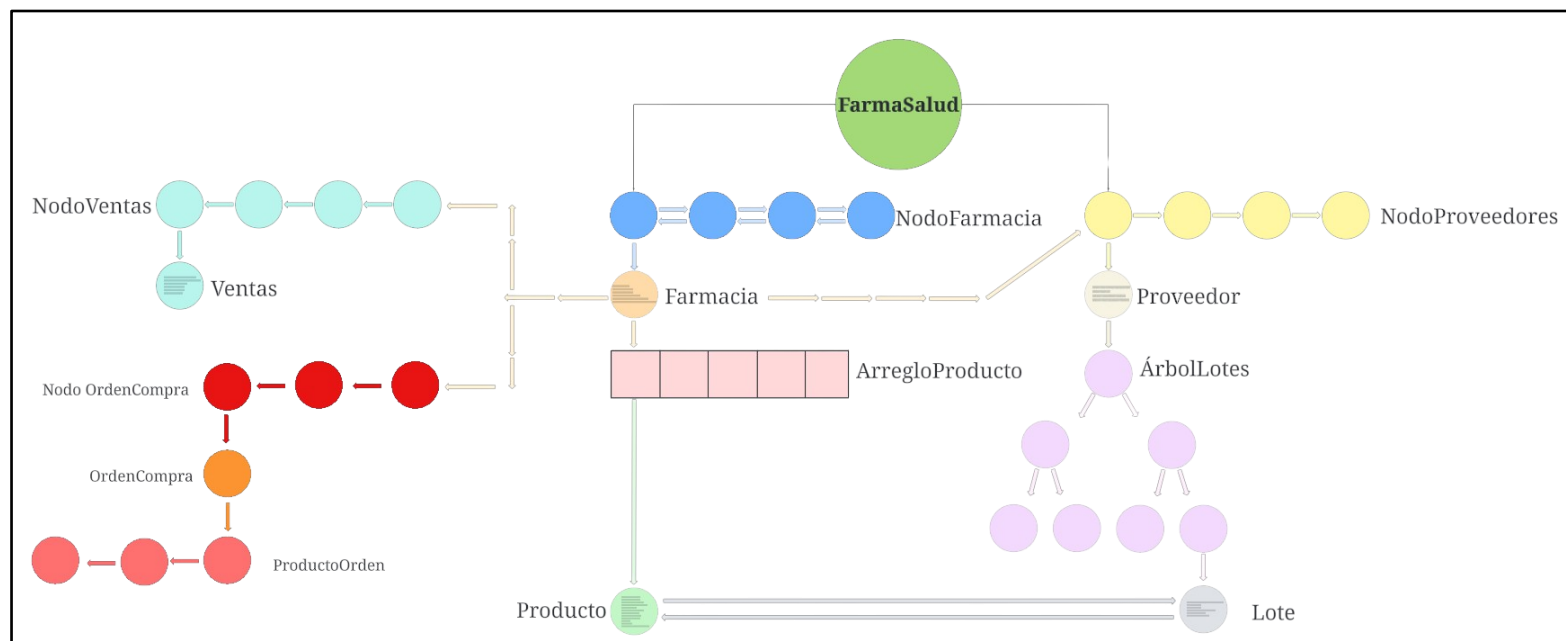
El programa registra todas las ventas realizadas, actualizando automáticamente el inventario de productos. Los detalles de cada venta, incluyendo el código del producto

vendido, cantidad, precio y cliente, se almacenan para futuras referencias. Además, el sistema genera informes y análisis de datos avanzados para identificar tendencias de ventas, patrones estacionales y productos de alta o baja rotación. Estos informes ayudan a optimizar el inventario y a tomar decisiones estratégicas informadas.

Diagramas y modelos explicativos:

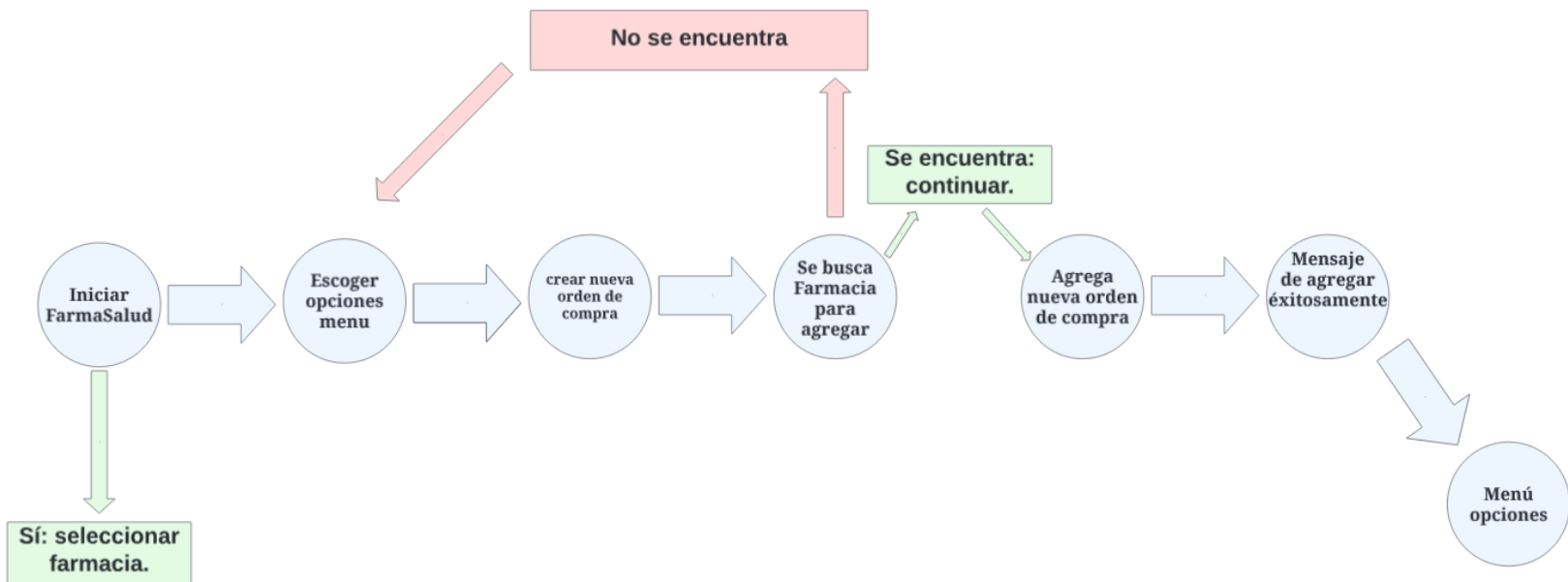
Se incluirán los siguientes diagramas, desarrollados en Lucidchart:

1. Diagrama de Estructura General: Organización de las estructuras de datos.



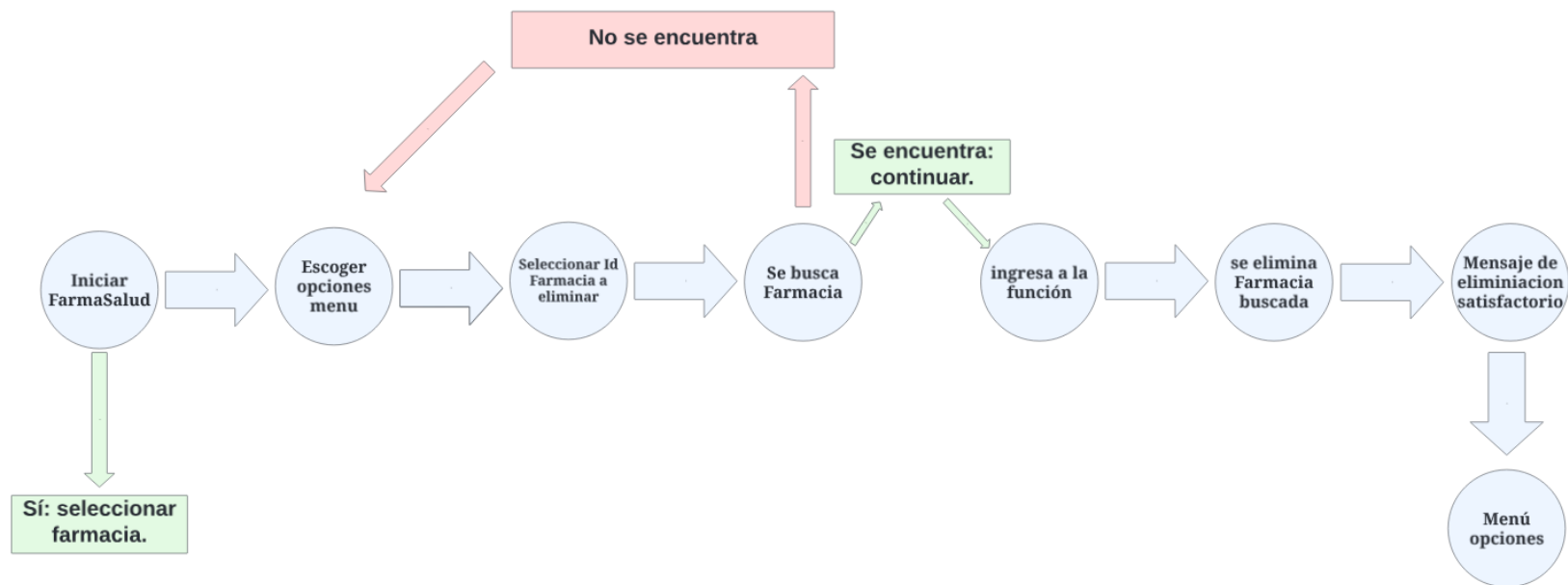
2. Diagrama de Flujo para Agregar Datos: Proceso para agregar datos en el último nivel de anidación.

En nuestro caso el último nivel de orden es el Orden de compra, entonces hicimos el diagrama en base a la agregación de este mismo:



3. Diagrama de Flujo para Eliminar Datos: Proceso para eliminar datos en el tercer nivel de anidación.

En nuestro caso el tercer nivel de orden es la farmacia, entonces hicimos el diagrama en base a la eliminación de esta misma:



Planificación

Carta Gantt grupal:

Grupo F.R.A.K.A.S.A.D.O.S.
Francisco Castro, Esteban Martínez, Juan Pablo Pizarro, Joaquín Saldivia



Inicio del proyecto:	vie, 24-5-2024	
mana para mostrar:	1	

[illegible]

El proyecto de desarrollo de la aplicación de gestión de inventarios para FarmaSalud fue desarrollada para resolver los desafíos asociados con la administración de una amplia red

de farmacias a lo largo del país. A través de un enfoque sistemático y el uso de diversas estructuras de datos en el lenguaje de programación C.

La implementación de estructuras de datos como listas enlazadas, tanto simples, dobles, árboles binarios de búsqueda y arreglos, ha permitido un manejo eficiente del inventario. Estas estructuras facilitan la búsqueda rápida, inserción y eliminación de productos y proveedores, lo cual es crucial para la operatividad diaria y la capacidad de respuesta de las farmacias.

La decisión de crear un campo específico para identificar productos que requieren receta ha simplificado el control, la venta y el registro de medicamentos regulados. Esto asegura que FarmaSalud cumpla con todas las normativas, garantizando la seguridad de los clientes.

La capacidad de generar alertas para productos próximos a caducar y para aquellos con niveles bajos de stock es un componente crítico de la aplicación. Estas alertas ayudan a prevenir pérdidas económicas por productos caducados y aseguran la disponibilidad continua de productos esenciales, mejorando la gestión y planificación del inventario.

El hecho de modularizar el proyecto es una buena práctica para el futuro para mejorar sus posibles reforzamientos y actualizaciones, integrando nuevas funcionalidades sin comprometer la estabilidad del sistema. La claridad y organización del código también contribuyen a un mantenimiento más sencillo y eficiente.

La implementación de un menú interactivo para la gestión de inventarios, ventas, proveedores y órdenes de compra mejora significativamente la experiencia del usuario. Esta interfaz facilita la operación del sistema, permitiendo a los usuarios realizar tareas de gestión de manera rápida y eficiente, y contribuyendo a una mejor toma de decisiones.

La capacidad de generar reportes y realizar análisis de datos permite identificar tendencias de ventas y patrones estacionales. Este análisis es fundamental para la toma de decisiones estratégicas, optimizando el reabastecimiento y la distribución de productos en las distintas farmacias.

La aplicación también incluye funciones específicas para la distribución óptima de productos entre las farmacias, asegurando que cada una tenga el stock adecuado de los productos más demandados según su ubicación y clientela. Este proceso requiere un análisis continuo de datos de ventas y una distribución eficiente de productos entre las farmacias, asegurando una respuesta rápida y efectiva a la demanda.

El desarrollo del proyecto de gestión de inventarios para FarmaSalud representa un avance significativo en la optimización de los procesos de esta y en la mejora de la eficiencia. La aplicación no solo permite una gestión precisa del inventario, sino que también proporciona herramientas el control de productos, de stock, y la toma de decisiones basada en datos y alertas. En conjunto, estas mejoras contribuyen a una mejor sostenibilidad a largo plazo para FarmaSalud.

Anexo

El anexo debe incluir el código fuente y los currículos de cada uno de los integrantes del grupo en formato estandarizado.

Carta Gantt:

[Carta Gantt FRAKASADOS](#)

Curriculums:

[Curriculums FRAKASADOS](#)

<https://drive.google.com/drive/u/1/folders/1fkiJBpIdJs2Qs6ntohzpyXzqsFuj9-MH>

Código Fuente:

<https://replit.com/@fravnsaavedra/Codigo-FarmaSalud-Grupo-FRAKASADOS#main.c>

Herramientas de tecnología avanzadas utilizadas:

Open AI - Chat GPT.

<https://openai.com/es-US/>