Interfaces Gráficas de Usuario

PROYECTO CONTADOR CALORIAS

Javier García Pechero

javigp@usal.es

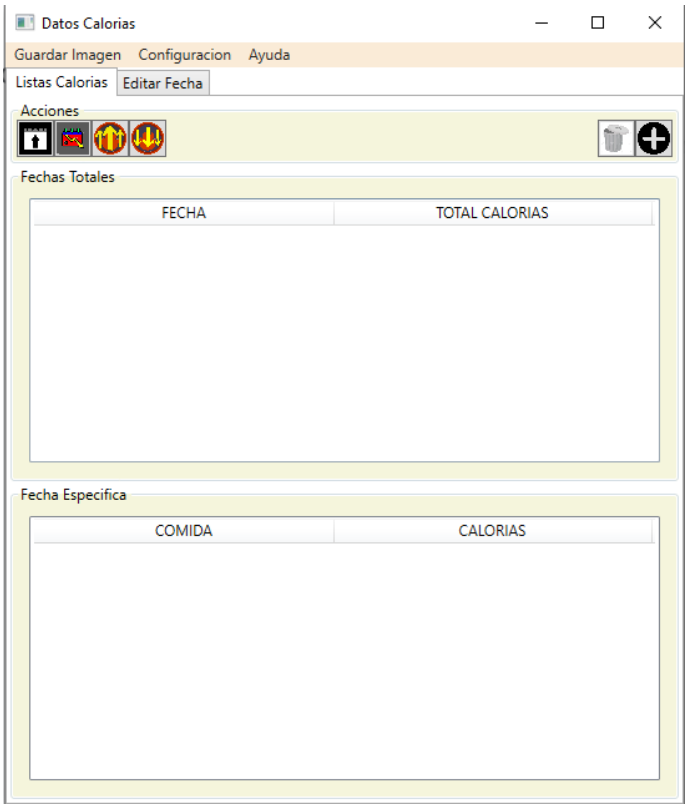
Curso 2021-2022

ÍNDICE

MANUAL DE USUARIO

Iniciamos el programa y observamos dos ventanas:

1. Gráficos Ingesta Calorias (Ventana Principal)



2. Datos Calorias (Ventana Secundaria)

VENTANA 1

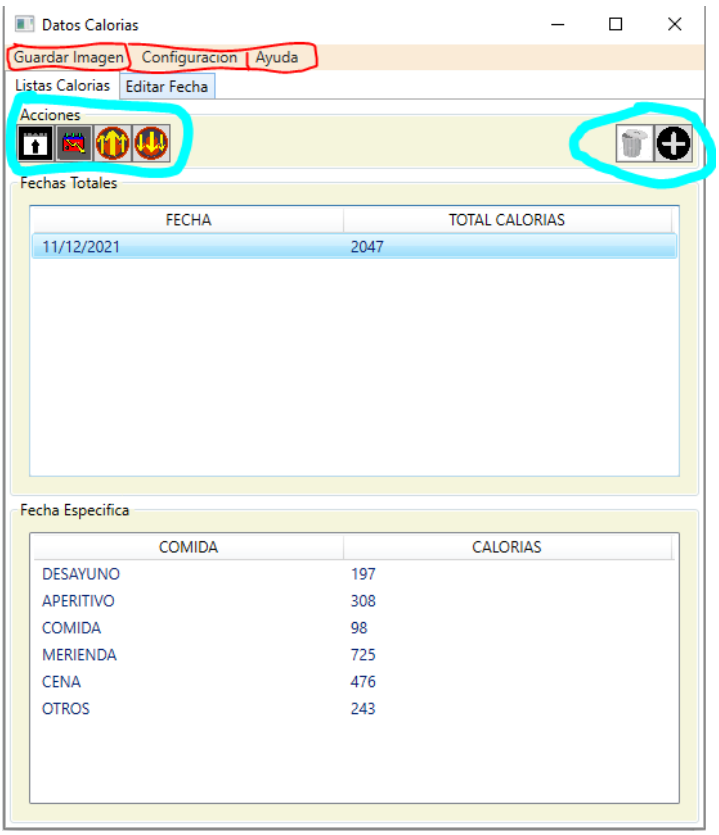
Mostrará las gráficas con los datos correspondientes que vayamos introduciendo. A su vez cuenta con dos botones que aparecen si tenemos el ratón sobre la ventana. El primer botón (Símbolo calorías azulado) abre la **ventana 2**, por defecto se abre sola al iniciar el programa, pero si se cierra, este botón habilita su despliegue. El botón 2 (Gráficas) resetea la **ventana 1** si está mostrando las gráficas de una fecha concreta para volver a mostrar la gráfica global.

El número 3 no es un botón pero indica las calorías exactas según la posición del ratón. Cuando se visualicen gráficas puede servir de ayuda para ver con exactitud las calorías.

VENTANA 2

Apartado Listas Calorías:

Visualmente observamos dos List Views, uno para las fechas totales y otro para una fecha específica. El primero guarda todas las fechas que vayamos introduciendo. El segundo solamente aparecerán datos una vez que seleccionemos una fecha en el primer List View, para que muestre cada comida de forma específica. Para dejar de seleccionar, simplemente pulsando en el propio ListView 1 o cualquier botón de la ventana o dando al botón reset de la **ventana 1.** A continuación se explicará los botones (rodeados en azul).



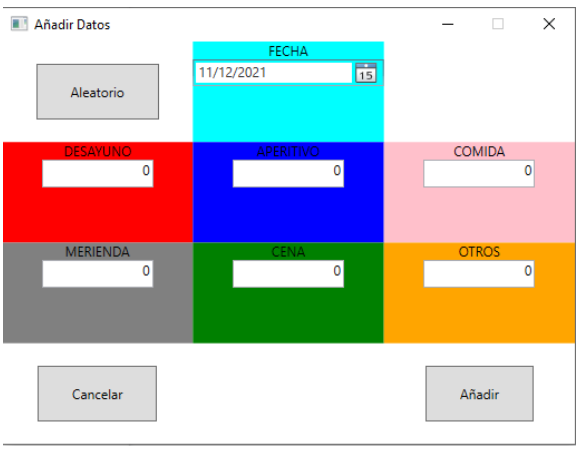
Cada uno de los botones repercuten a la hora de representarse en el ListView 1 (Fechas totales), así como en la **ventana 1** en la representación de la gráfica**.**

De izquierda a derecha:

1. Ordena por fecha ascendente
2. Ordena por fecha descendente
3. Ordena por calorías ascendente
4. Ordena por calorías descendente
5. Borra todas las fechas
6. Añadir fecha

Para borrar únicamente una fecha, selecciona la fecha deseada con click izquierdo y posteriormente haga click derecho sobre ella, le mostrará un mensaje si desea borrar la fecha seleccionada.

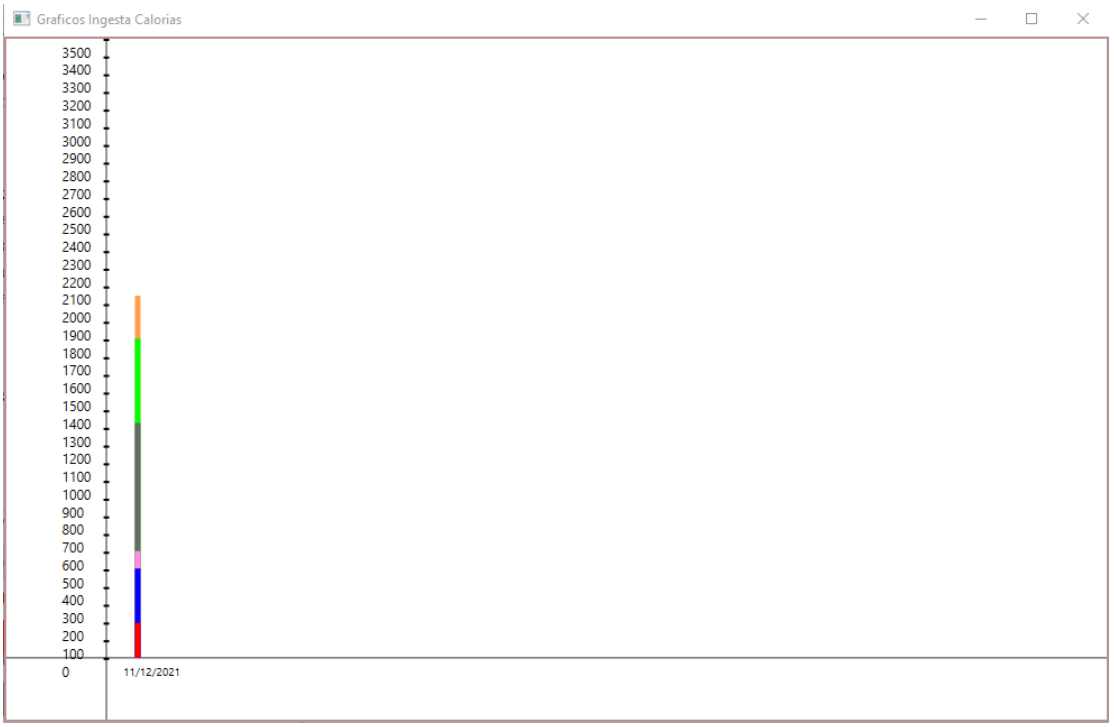
En cuanto al botón 6 añadir, al pulsarlo se mostrará la siguiente interfaz:



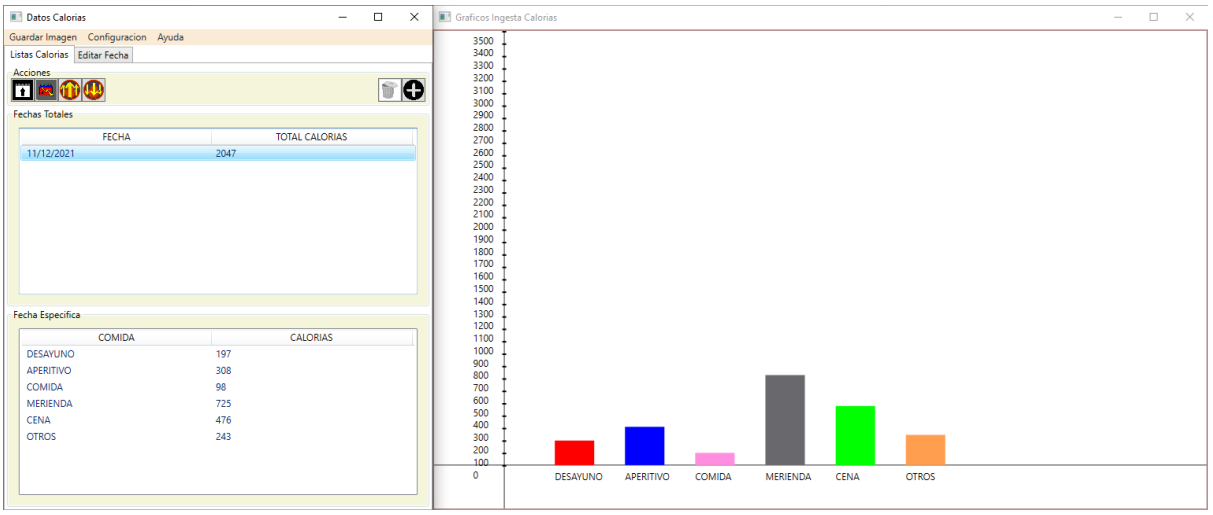
Una interfaz muy simple que indica por código de colores y título cada una de las comidas del día. Cuenta con un DatePicker para seleccionar la fecha y un botón aleatorio que rellena los datos de las comidas aleatoriamente.

Para finalizar el diálogo con la ventana cuenta con el botón cancelar que cierra la ventana y no provoca cambios. Y el botón añadir que todos los valores introducidos se verán reflejados tanto en la gráfica global como en el ListView de Fechas Totales. Si se introduce una fecha ya introducida en la base de datos dará error, así como si alguno de los campos de alguna comida está vacío de datos.

Ejemplo sin seleccionar ListView 1(Fechas Totales)



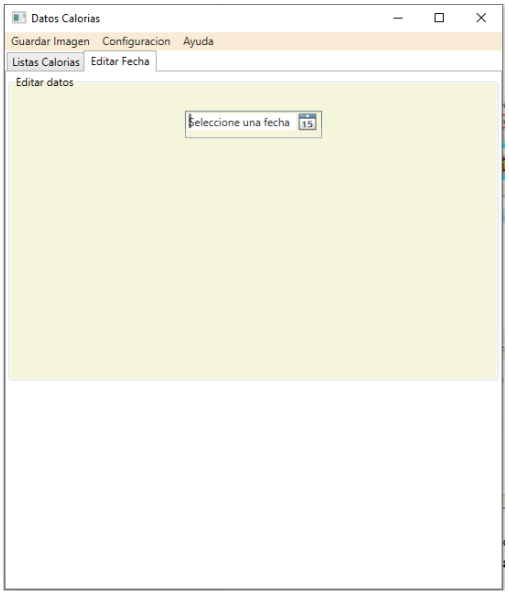
Ejemplo seleccionando fecha en ListView 1 (Fechas Totales)



Apartado Editar Fecha:

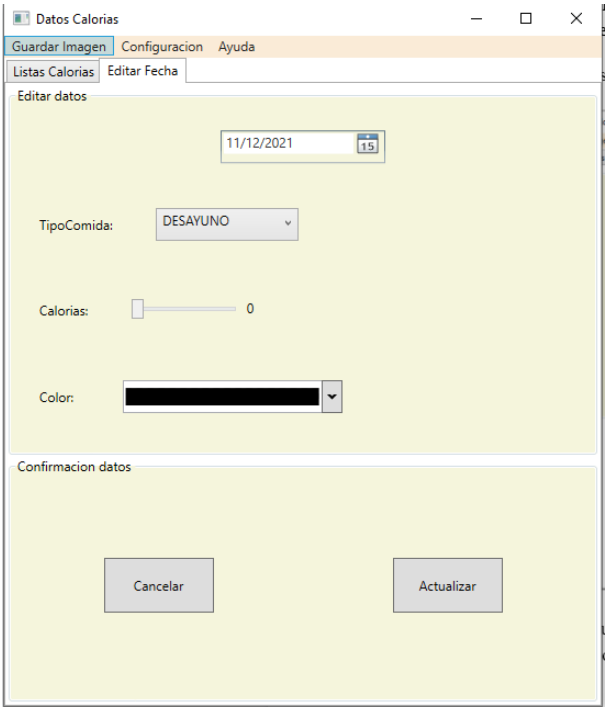
Este segundo apartado nos sirve para editar una fecha concreta, **solamente se habilitará si existen fechas en la base de datos y la fecha introducida es una de las que existan.**

Al seleccionarlo observamos lo siguiente:

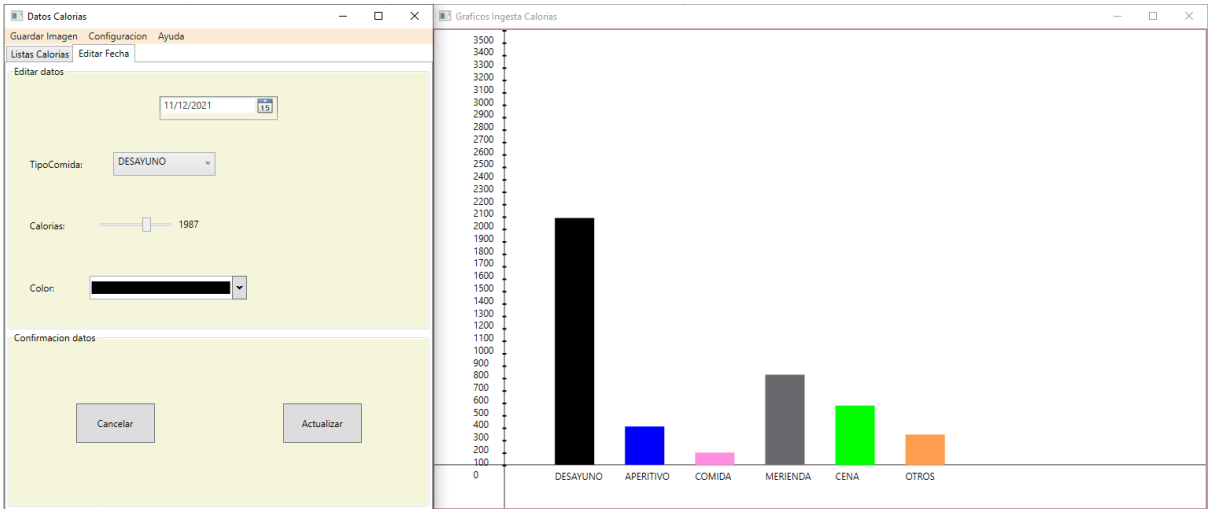


Si introducimos una fecha que no existe en la base de datos o es incorrecta, dará error y nos llevará al apartado Lista Calorías.

Si introducimos una fecha correcta observaremos la siguiente interfaz:



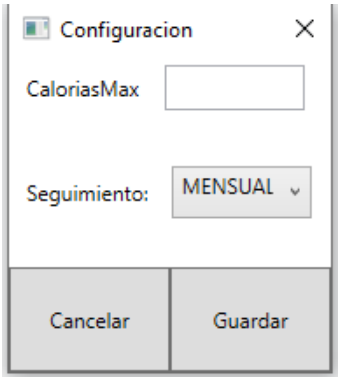
Un apartado TipoComida que nos permite seleccionar qué comida del día queremos editar. Un slider para introducir las calorías que deseamos que haya de esa comida. Y un apartado color que nos abrirá un desplegable con colores para seleccionar el que queramos. Para aplicar cambios solo basta en dar al botón actualizar, al pulsarlo la **ventana 1** observaremos los cambios que se han producido, pues mostrará la gráfica de la fecha concreta desglosada en comidas y podremos realizar los cambios inmediatamente tras pulsar el botón.



Los Menu Items de arriba del todo de la **ventana 2**:

El primero Guardar Imagen nos permite guardar una imagen de la **ventana 1**, se guarda en función del tamaño que tenga dicha ventana en el momento de aceptar el guardado. Se recomienda guardarla teniendo la ventana a pantalla completa.

El botón configuración al pulsarlo nos mostrará una ventana nueva:



Nos permitirá introducir las calorías máximas que se podrán ver en la **ventana 1**, así como el máximo de gráficas en la opción global. La aplicación está pensada para hacer un seguimiento semanal (7 gráficas máx) o mensual (31 gráficas max) a partir de la primera fecha que marque la ordenación según las opciones dichas anteriormente. Cabe destacar que si no se selecciona ningún botón de ordenación las gráficas se van añadiendo consecutivamente independientemente de cantidad calorías o fecha.

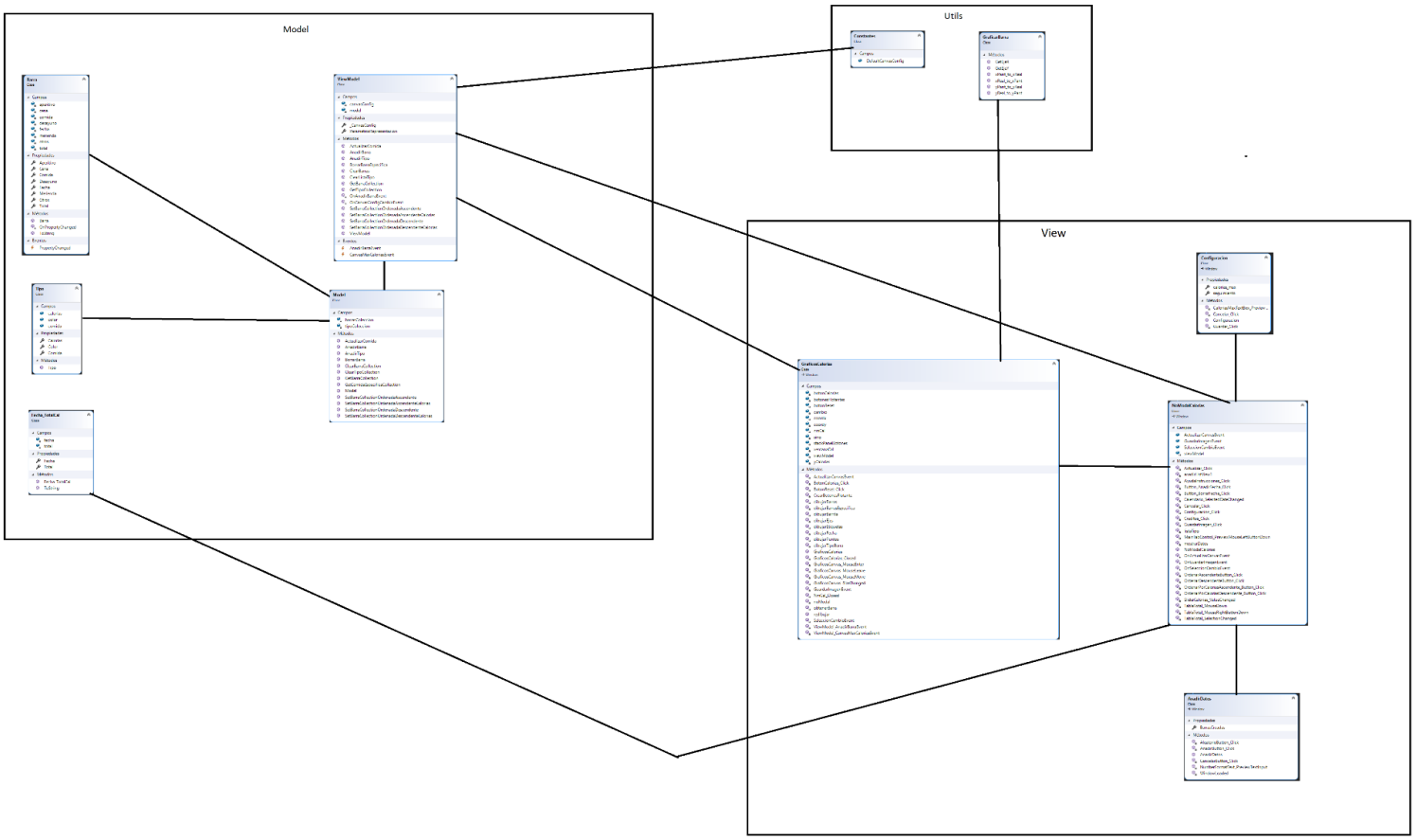
Finalmente el botón ayuda tiene dos opciones:

1.Instrucciones: Explica brevemente este mismo manual de usuario

2.Creditos: Muestra el autor

Manual de programador

Se explicará la estructura y funcionamiento del programa dividido por paquetes. Cabe destacar que la aplicación está basada en un VMMV. Este patrón facilita trabajar con WPF ayudando a la sincronización entre varias ventanas.



(Para mejor visualización véase diagrama_clases.png)

Modelo

Paquete: “Model”

Aquí encontramos la funcionalidad central de la aplicación. Está formada por 4 clases:

- **Model:** Interfaz que implementa el almacenamiento de los datos. Se trabaja con dos ObservableCollection. En dicha interfaz se implementan también las operaciones “Clear”, “Get” y “Add” de cada colección. Solamente para la colección Barra se introduce la operación “Delete”. Además cuenta con funciones que ordenan la ObservableCollection de barras en función del dato calorías o fecha. También existe

una función que actualiza los datos de una comida en específico, se manifiesta al editar una fecha concreta y la comida concreta.

- **ViewModel:** Interfaz que implementa la carga del Modelo y la carga de la configuración inicial de las coordenadas de los ejes del canvas. Sirve de puente de datos entre las clases que se encuentran en la vista y el modelo. Cuenta con dos Eventos que son lanzados al añadir una barra al modelo y al cambiar la configuración de la ventana principal. Estos dos eventos se explican en la clase **GraficosCalorias**.
- **Tipo:** Esta interfaz está formada por tres propiedades: calorías, comida y color. Se utiliza para definir los datos de cada comida específica.
- **Barra:** Interfaz que define las propiedades de cada fecha. Contiene propiedades de **Tipo** con cada comida, además de una propiedad fecha y total. Una ObservableCollection del modelo está formada por esta definición. Implementa la interfaz "INotifyPropertyChanged" pero no se llega a usar en el programa.
- **Fecha_TotalCal:** Interfaz que define propiedades para la visualización del ListView Fechas Totales.

Utils

Paquete: "Utils"

- **Constantes:** Interfaz que implementa la definición de las propiedades y la carga por defecto de los ejes del canvas.
- **GraficarBarra:** Interfaz que implementa las funciones de creación de los ejes X e Y. También cuenta con funciones para la obtención de posiciones de los puntos en el canvas.

Vista

Paquete: "View"

- **GraficosCalorias:** Esta interfaz implementa la ventana principal donde se mostrará los gráficos. Cuenta con todos los métodos para dibujar las barras, la creación de los botones flotantes y sus respectivos eventos al pulsarlos. También gestiona los eventos de entrada y salida del ratón en la ventana, haciendo que aparezcan y desaparezcan los botones flotantes. El otro botón sirve para resetear la gráfica de la ventana. También se gestiona el evento que gestiona el cambio de tamaño de la ventana para mostrar las gráficas correctamente de manera proporcional al tamaño de la ventana. Por último se gestiona el evento MouseMove para indicar la cantidad de calorías según la posición del ratón en la ventana.

  Se encarga de la creación de la ventana **No Modal**, que está enrutada con un botón flotante. En cuanto a la gestión de eventos procedentes de la ventana **NO MODAL** tenemos los siguientes:

Para evitar abrir varias ventanas No Modales, se ha gestionado el evento closed de la ventana **No Modal**. También gestiona el evento GuardarImagenEvent, guarda una imagen de cómo está la ventana en el momento de darle a guardar**.** El evento SeleccionCambioEvent se gestiona indicando que tiene que dibujar las gráficas específicas de un día concreto. Mientras que ActualizarCanvasEvent gestiona que se vuelva a indicar mostrar la gráfica global.

También cuenta con las llamadas de los eventos AnadirBarraEvent y CanvasMaxCaloriasEvent del viewModel asociados a la llamada del método redibujar().

- **NoModalCalorías:** En esta interfaz tenemos métodos para añadir datos,borrarlos y editarlos. Para añadir datos interviene el evento del botón AnadirFecha, se crea una nueva ventana modal la cual se explicará posteriormente, al concluir el diálogo con dicha ventana se invoca un evento para que muestre en la ventana de la gráfica la barra añadida. Al seleccionar un elemento de la ListView1 de fechas totales observamos cómo se invoca el OnSeleccionCambioEvent para que llame a la función mencionada anteriormente en **GraficosCalorias** y muestre la gráfica de las comidas específicas de una fecha. Siempre que se pulse un botón ya sea de ordenar,borrar o editar, se invoca un evento para comunicárselo a la ventana **GraficosCalorias** y se actualice**,** además se comunica con el viewModel para el guardado de datos de cualquier cambio realizado.

  También cuenta con un botón configuración que abre una ventana modal que al finalizar el diálogo con ella, comunica los parámetros del canvas.

- **AnadirDatos:** Esta interfaz está formada por métodos como una expresión regular para evitar introducir caracteres no deseados en las textbox. Cuenta con el tratado al pulsar el boton anadir y cancelar. Además cuenta con un botón aleatorio para el autorellenado de datos. También cuenta con un método de autocarga que al entrar tiene seleccionado la fecha del día actual en el que se está trabajando con el programa y las textBox cuentan con valor 0.
- **Configuración:** Esta interfaz sirve para introducir los datos para cambiar la configuración del canvas, que eso ocurre en **NoModalCalorias.**

BIBLIOGRAFÍA

En el código hay bibliografía implementada, esto es la mayoría de sitios que guardé pero seguramente falten muchos más.

https://www.c-sharpcorner.com/UploadFile/mahesh/working-with-datetime-using-C-Sharp/

https://wpf-tutorial.com/es/497/controles-adicionales/the-datepicker-control/

https://stackoverflow.com/questions/9301030/how-to-get-and-bind-the-value-of-slider-in-c-sharp-wpf

https://stackoverflow.com/questions/1268552/how-do-i-get-a-textbox-to-only-accept-numeric-input-in-wpf

http://tutorialgenius.blogspot.com/2014/12/saving-window-or-canvas-as-png-bitmap.html

https://stackoverflow.com/questions/7929646/how-to-programmatically-select-a-tabitem-in-wpf-tabcontrol*/

https://thedeveloperblog.com/datepicker

https://www.c-sharpcorner.com/blogs/wpf-color-picker-c-sharp

https://www.telerik.com/forums/how-to-unselect-an-item

https://htmlcolorcodes.com/es/

https://www.generacodice.com/en/articolo/236981/wpf-coi-visual-especificado-ya-es-un-hijo-de-otro-visual-o-la-ra-z-de-un-compositiontarget