En vuestra clase "nave" creamos 2 propiedades una que será la posición en el canvas de la nave ya sea X para moverse horizontal o Y vertical, y la otra será la velocidad a la que se moverá la nave en el canvas.

En en el ViewModel usaremos la clase DispatcherTimer.

private DispatcherTimer dispatcherTimer { get; set; }

public MainPageVM() //constructor

{

dispatcherTimer = new DispatcherTimer();

dispatcherTimer.Interval = new TimeSpan(0, 0, 0, 0, 1); //para comprobar cada milisegundo si se esta pulsando alguna tecla

dispatcherTimer.Tick += timerTick; //para que cada milisegundo que se comprueba realice alguna funcion en este caso timerTick

moviendoX = false;

moviendoY = false;

\_nave = new Nave(500,500,0);

}

private void timerTick(object sender, object e)

{

move();

}

Con cada tick del intervalo del dispatcherTimer se llama a move() y este método lo que hace es que suma la posición(X o Y) mas la velocidad (esta la tendríamos si pulsáramos alguna tecla)

public void move()

{

Double posicionFutura;

if (moviendoY) {

posicionFutura = \_nave.posY + \_nave.velocidad;

if (posicionFutura > 0 && posicionFutura < 500) {

\_nave.posY += \_nave.velocidad;

}

}

if (moviendoX) {

posicionFutura = \_nave.posX + \_nave.velocidad;

if (posicionFutura > 0 && posicionFutura < 1180)

{

\_nave.posX += \_nave.velocidad;

}

}

NotifyPropertyChanged("nave");

}

Para comprobar el evento cuando pulsamos alguna tecla creamos un metodo que compruebe que tecla se pulsa y si se pulsa que realice la funcion que queramos

public void Grid\_KeyDown(object sender, KeyRoutedEventArgs e) //Grid\_KeyDown es el nombre que yo le he dado se puede llamar como querais

{

if (e.Key == VirtualKey.Up) //movimiento flecha arriba

{

arriba();

dispatcherTimer.Start(); //empieza el dispatcherTimer

moviendoY = true;

moviendoX = false;

}

if (e.Key == VirtualKey.Down)//movimiento flecha abajo

{

abajo();

dispatcherTimer.Start();

moviendoY = true;

moviendoX = false;

}

if (e.Key == VirtualKey.Right) //movimiento flecha derecha

{

derecha();

dispatcherTimer.Start();

moviendoX = true;

moviendoY = false;

}

if (e.Key == VirtualKey.Left)//movimiento flecha izquierda

{

izquierda();

dispatcherTimer.Start();

moviendoX = true;

moviendoY = false;

}

}

Si dejamos de pulsar la tecla tendra que realizar una funcion que pare el dispatcherTimer

public void Grid\_KeyUp(object sender, KeyRoutedEventArgs e)

{

if (e.Key == VirtualKey.Up || e.Key == VirtualKey.Down || e.Key == VirtualKey.Left || e.Key == VirtualKey.Right) {

dispatcherTimer.Stop();

moviendoY = false;

moviendoX = false;

}

}

Ahora para que se realice el movimiento tenemos que realizar un bindeo para ello utilizaremos x:Bind

Para usar x:Bind iremos al codigo behind de la vista en mi caso MainPage.xaml.cs

public sealed partial class MainPage : Page

{

public MainPageVM vMGame { get; }

public MainPage()

{

this.InitializeComponent();

vMGame = (MainPageVM)this.DataContext;

}

}

Una vez tenemos esto vamos a MainPage.xaml y realizamos el bindeo del evento, para ello usaremos la propiedades keydown y keyup

<Image Canvas.ZIndex="20" Name="stkJugador1" Width="200" Height="200" Source="ms-appx:///Assets/Spaceship-PNG-File.png"

KeyDown="{x:Bind vMGame.Grid\_KeyDown,Mode=OneWay}" KeyUp="{x:Bind vMGame.Grid\_KeyUp}"

Canvas.Left="{Binding nave.posX, Mode=TwoWay, UpdateSourceTrigger=PropertyChanged}"

Canvas.Top="{Binding nave.posY, Mode=TwoWay, UpdateSourceTrigger=PropertyChanged}" ></Image>

Ya solo falta el paso final, si haceis todo lo anterior vereis que no funciona

Para ello debemos poner preparar la interaccion con el panel, asique en el mismo codigo behind haremos esta función:

private void allowfocus\_Loaded(object sender, RoutedEventArgs e)

{

Window.Current.Content.KeyDown += this.vMGame.Grid\_KeyDown;

Window.Current.Content.KeyUp += this.vMGame.Grid\_KeyUp;

}

Ahora realizamos el bindeo en el panel que queremos que se produzca el movimiento de la nave tenemos que usar la propiedad Loaded

<Canvas x:Name="canvas" Background="Black" Loaded="allowfocus\_Loaded" MaxHeight="1000" MaxWidth="1500">

......

</Canvas>

Aqui abajo os dejo las funciones para los movimientos

public void abajo()

{

if (\_nave.posY < 500)

{

\_nave.velocidad = 10;

}else{

\_nave.velocidad = 0;

}

}

public void arriba()

{

if (\_nave.posY > 0 && \_nave.posY - 10 > 0)

{

\_nave.velocidad = -10;

}

else{

\_nave.velocidad = 0;

}

}

public void derecha()

{

if (\_nave.posX < 1180) {

\_nave.velocidad = 10;

}else{

\_nave.velocidad = 0;

}

}

public void izquierda()

{

if (\_nave.posX > 0 && \_nave.posX - 10 > 0) {

\_nave.velocidad = -10;

}else{

\_nave.velocidad = 0;

}

}