



MANUAL TECNICO

MANUAL DE USUARIO

HERRAMIENTAS MULTIMEDIA

UNIVERSIDAD POLITECNICA DE VICTORIA

OSCAR FRANCISCO FLORES GALLEGOS



1.- INDICE

INTRODUCCION.....	2
DESARROLLO.....	3-15
CONCLUSION.....	15

2.- INTRODUCCION

El propósito de este manual técnico es el de explicar la forma en que se realizó la interfaz de la galería de recopilación de evidencias. Así mismo se dará detalle de cómo es que funciona la misma y los aspectos básicos que permiten la realización de las animaciones de la interfaz.

La interfaz sigue un diseño el cual permite realizar series de Tweens para generar la “animación” de giro de un cubo, aunque esta no es 100% en 3D, se puede apreciar el giro como si se estuviera viendo un cubo de frente y lo rotáramos hasta la cara contraria a la que estamos viendo.

Solo se mostraran las secciones de código que tengan que ver con la animación de movimiento de la interfaz, además de que solo se hablara sobre este tema a lo largo de este manual.

3.- DESAROLLO

A continuación se explicara el código necesario para el funcionamiento de la interfaz.

Lo primero que tenemos a parte son las variables, las dos primeras solo son para cargar los archivos de las practicas, el resto son variables para el funcionamiento de la interfaz.

En la función principal tenemos declarados los Listener de los botones del menú, el cual está presente en todos los frames de operación. También se invoca una función llamada Conclusión, la cual tiene como objetivo una pequeña animación que se verá un poco más adelante. Además se encuentran los tweens de la portada.

En la siguiente imagen se aprecia el código de creación de un Sprite el, siendo en este donde está ubicada la sección de los Videos de las conclusiones en Inglés y Español.

```
var urlDes:URLRequest;
var urlFile:URLLoader;
var fondo2:Fondo2;
var extenderImagen:Sprite;
var transicion:int := 0;
var botonCerrar:BotonCerrar := new BotonCerrar();
var botonCerrar2:BotonCerrar := new BotonCerrar();
var boton1:BotonEspañol := new BotonEspañol();
var boton2:BotonIngles := new BotonIngles();
var frameActual:int := 1;
var v:simpleFlv := new simpleFlv();
var mov1:Tween;
var mov2:Tween;

//Funcion Inicial
public function Main() {
    » Conclusion();
    » var mov11:Tween := new Tween(linea1, "x", Bounce.easeOut, 570, 157.40, 2, true);
    » var mov12:Tween := new Tween(linea2, "x", Elastic.easeOut, 570, 162.75, 2, true);
    » var mov13:Tween := new Tween(linea3, "x", Bounce.easeOut, 570, 183.25, 2, true);
    » var mov14:Tween := new Tween(linea4, "x", Elastic.easeOut, 570, 200.30, 2, true);
    » var mov15:Tween := new Tween(logos_mc, "y", Elastic.easeOut, -100, -5.95, 2, true);
    »
    » aro.practicas.addEventListener(MouseEvent.CLICK, Anim1);
    » aro.juegos.addEventListener(MouseEvent.CLICK, Anim4);
    » aro.mediaa.addEventListener(MouseEvent.CLICK, Anim3);
    » aro.tarea.addEventListener(MouseEvent.CLICK, Anim2);
    » videos.addEventListener(MouseEvent.CLICK, GoVideos);
```

```
» » » extenderImagen := new Sprite();
» » » extenderImagen.graphics.beginFill(0x005D3D, .75);
» » » extenderImagen.graphics.drawRect(-500, -50, 500, 300);
» » » addChild(extenderImagen);
» » }
```

En la siguiente imagen se aprecia la función Conclusión, la cual invocamos desde el inicio. Su propósito es simple, inicia un timer infinito el cual provoca la rotación de la flecha curvada que sale del botón de conclusión, como se aprecia en la siguiente imagen.

```
var timer:Timer = new Timer(10, contF++);
var contF:int = 0;

public function Conclusion() {
    >> timer.start();
    >> timer.addEventListener(TimerEvent.TIMER, tiempo);
}

//Funcion para que giren los molinos
function tiempo(e:TimerEvent):void{
    >> r1.rotation += 1;
}
```



En la siguiente sección se tiene las funciones de las animaciones iniciales (Anim), las cuales se invocan al presionar uno de los botones del menú. Todas estas funciones siguen el mismo patrón, crean un nuevo fondo a partir de la biblioteca y lo posicionan en la posición contraria a la que se desplazara y sin separación del fondo.

Después se hace que el fondo y el aro del menú se muevan hacia algún lado dependiendo de qué botón se presionó (practicar = arriba, tareas = derecha, multimedia = izquierda y juegos = abajo), y seguido de este se mueve también el segundo fondo creado. Cuando este acaba su movimiento, es decir, queda ubicado en el escenario, invoca su respectiva función Go para continuar con la animación en el siguiente frame.

El cambio entre cada una de estas funciones son los parámetros iniciales del segundo fondo y los parámetros finales del primer fondo y el aro. También se hacen invisibles los ítems del botón de la sección de Videos.

```

//ANIMACIONES
//PRACTICAS
private function Anim1(event:MouseEvent){
    >> fondo2 := new Fondo2 ();
    >>
    >> fondo2.x := 0;
    >> fondo2.y := 400;
    >> addChild(fondo2);
    >> mov1 := new Tween(fondo, "y", Regular.easeOut, fondo.y, -400, 2, true);
    >> mov1 := new Tween(aro, "y", Regular.easeOut, fondo.y, -400, 2, true);
    >> mov2 := new Tween(fondo2, "y", Regular.easeOut, fondo2.y, 0, 2, true);
    >>
    >> mov2.addEventListener(TweenEvent.MOTION_FINISH, GoPracticas);
    >>
    >> con.visible := false;
    >> r1.visible := false;
    >> videos.visible := false;
    >>
    >> switch(frameActual){
    >> >> case 2: pan1.visible := false; break;
    >> >> case 3: pan4.visible := false; break;
    >> >> case 4: pan3.visible := false; break;
    >> >> case 5: pan2.visible := false; break;
    >> }
}

```

```

//TAREAS
private function Anim2(event:MouseEvent){
    >> fondo2 := new Fondo2 ();
    >>
    >> fondo2.x := 550;
    >> fondo2.y := 0;
    >> addChild(fondo2);
    >> mov1 := new Tween(fondo, "x", Regular.easeOut, fondo.x, -550, 2, true);
    >> mov1 := new Tween(aro, "x", Regular.easeOut, aro.x, -550, 2, true);
    >> mov2 := new Tween(fondo2, "x", Regular.easeOut, fondo2.x, 0, 2, true);
    >>
    >> mov2.addEventListener(TweenEvent.MOTION_FINISH, GoTareas);
    >>
    >> con.visible := false;
    >> r1.visible := false;
    >> videos.visible := false;
    >>
    >> switch(frameActual){
    >> >> case 2: pan1.visible := false; break;
    >> >> case 3: pan4.visible := false; break;
    >> >> case 4: pan3.visible := false; break;
    >> >> case 5: pan2.visible := false; break;
    >> }
}

```

```

//MEDIA
private function Anim3(event:MouseEvent){
    >> fondo2 := new Fondo2;
    >>
    >> fondo2.x := -550;
    >> fondo2.y := 0;
    >> addChild(fondo2);
    >> mov1 := new Tween(fondo, "x", Regular.easeOut, fondo.x, -550, 2, true);
    >> mov1 := new Tween(aero, "x", Regular.easeOut, aero.x, -550, 2, true);
    >> mov2 := new Tween(fondo2, "x", Regular.easeOut, fondo2.x, 0, 2, true);
    >>
    >> mov2.addEventListener(TweenEvent.MOTION_FINISH, GoMedia);
    >>
    >> con.visible := false;
    >> r1.visible := false;
    >> videos.visible := false;
    >>
    >> switch(frameActual){
    >> >> case 2: pan1.visible := false; break;
    >> >> case 3: pan4.visible := false; break;
    >> >> case 4: pan3.visible := false; break;
    >> >> case 5: pan2.visible := false; break;
    >> }
}

```

```

//JUEGOS
private function Anim4(event:MouseEvent){
    >> fondo2 := new Fondo2 ();
    >>
    >> fondo2.x := 0;
    >> fondo2.y := -400;
    >> addChild(fondo2);
    >> mov1 := new Tween(fondo, "y", Regular.easeOut, fondo.y, -400, 2, true);
    >> mov1 := new Tween(aero, "y", Regular.easeOut, fondo.y, -400, 2, true);
    >> mov2 := new Tween(fondo2, "y", Regular.easeOut, fondo2.y, 0, 2, true);
    >>
    >> mov2.addEventListener(TweenEvent.MOTION_FINISH, GoJuegos);
    >>
    >> con.visible := false;
    >> r1.visible := false;
    >> videos.visible := false;
    >>
    >> switch(frameActual){
    >> >> case 2: pan1.visible := false; break;
    >> >> case 3: pan4.visible := false; break;
    >> >> case 4: pan3.visible := false; break;
    >> >> case 5: pan2.visible := false; break;
    >> }
}

```

Lo siguiente es el bloque de funciones Go, el cual tiene como objetivo ser la continuación de la animación de la interfaz en el nuevo frame. Cuando una función Anim acaba, el segundo fondo queda sobre el escenario. Cuando se invoca la función Go, se realiza algo similar a lo que se hizo para que ocurriera la animación inicial. Se hace que el segundo fondo siga el camino que tomo el primer fondo, después se cambia al siguiente frame, lo cual hace que la variable indicadora frameActual tome el valor del frame correspondiente, la función de esta variables se verá más adelante.

Ya habiendo cambiado de frame se posiciona el primer fondo el aro y el conjunto de botones de ese frame (pan) en la posición inicial del segundo fondo, para luego desplazarse de regreso al escenario, y activar los Listener de cada botón, indicando a que grupo pertenecen. La variable transición es una variable que cambia cuando se sale de la portada, y es solo para indicar al programa que ya no se está en la portada. Para mayor efecto el grupo pan de botones se pone como invisible y se visible al llamar esa función, aunque esto solo tiene repercusión cuando se cambia de un menú a otro, y no desde la portada. Al final se vuelven a hacer visibles los ítems del botón de la sección de los Videos.

En todas las funciones Go se aplica la misma lógica, solo cambiando los parámetros, como lo que sucedía con las funciones Anim.

```
//Ir al frame 2, Practicas
private function GoPracticas(event:TweenEvent){
    >> mov2 := new Tween(fondo2, "y", Regular.easeOut, fondo2.y, -400, 2, true);
    >> gotoAndStop(2);
    >> frameActual := 2;
    >>
    >> fondo.y := 400;
    >> aro.y := 400;
    >> pan1.y := 400;
    >> mov2 := new Tween(aros, "y", Regular.easeOut, aro.y, 0, 2, true);
    >> mov1 := new Tween(fondo, "y", Regular.easeOut, fondo.y, 0, 2, true);
    >> mov1 := new Tween(pan1, "y", Regular.easeOut, pan1.y, 51.50, 2, true);
    >>
    >> pan1.p1.addEventListener(MouseEvent.CLICK, P1);
    >> pan1.p2.addEventListener(MouseEvent.CLICK, P2);
    >> pan1.p3_1.addEventListener(MouseEvent.CLICK, P3_1);
    >> pan1.p3_2.addEventListener(MouseEvent.CLICK, P3_2);
    >> pan1.p4.addEventListener(MouseEvent.CLICK, P4);
    >> pan1.p5_1.addEventListener(MouseEvent.CLICK, P5_1);
    >> pan1.p5_2.addEventListener(MouseEvent.CLICK, P5_2);
    >> pan1.p6.addEventListener(MouseEvent.CLICK, P6);
    >> pan1.p8.addEventListener(MouseEvent.CLICK, P8);
    >> pan1.p9.addEventListener(MouseEvent.CLICK, P9);
    >> pan1.p11.addEventListener(MouseEvent.CLICK, P11);
    >>
```



```

»   if(transicion==0){
»       transicion=1;
»   }
»
»   pan1.visible=true;
»   mov1.addEventListener(TweenEvent.MOTION_FINISH, BtnConclusion);
»   }

```

```

//Ir al frame 3, -Juegos
private function GoJuegos(event:TweenEvent){
»   mov2=new Tween(fondo2, "y", Regular.easeOut, fondo2.y, 400, 2, true);
»   gotoAndStop(3);
»   frameActual=3;
»
»   fondo.y=-400;
»   aro.y=-400;
»   pan4.y=-400;
»   mov2=new Tween(aro, "y", Regular.easeOut, aro.y, 0, 2, true);
»   mov1=new Tween(fondo, "y", Regular.easeOut, fondo.y, 0, 2, true);
»   mov1=new Tween(pan4, "y", Regular.easeOut, pan4.y, 61.85, 2, true);
»
»   pan4.j1.addEventListener(MouseEvent.CLICK, J1);
»   pan4.j2.addEventListener(MouseEvent.CLICK, J2);
»
»   if(transicion==0){
»       transicion=1;
»   }
»
»   pan4.visible=true;
»   mov1.addEventListener(TweenEvent.MOTION_FINISH, BtnConclusion);
»   }

```

```

//Ir al frame 4, Multimedia
private function GoMedia(event:TweenEvent){
    >> mov2 := new Tween(fondo2, "x", Regular.easeOut, fondo2.x, -550, 2, true);
    >> gotoAndStop(4);
    >> frameActual := 4;
    >>
    >> fondo.x := -550;
    >> aro.x := -550;
    >> pan3.x := -550;
    >> mov2 := new Tween(aro, "x", Regular.easeOut, aro.x, 75, 2, true);
    >> mov1 := new Tween(fondo, "x", Regular.easeOut, fondo.x, 0, 2, true);
    >> mov1 := new Tween(pan3, "x", Regular.easeOut, pan3.x, 173.8, 2, true);
    >>
    >> pan3.m1.addEventListener(MouseEvent.CLICK, M1);
    >> pan3.m2.addEventListener(MouseEvent.CLICK, M2);
    >> pan3.m3.addEventListener(MouseEvent.CLICK, M3);
    >>
    >> if(transition == 0){
    >>     transition := 1;
    >> }
    >>
    >> pan3.visible := true;
    >> mov1.addEventListener(TweenEvent.MOTION_FINISH, BtnConclusion);
    >> }

```

```

//Ir al frame 5, Tareas
private function GoTareas(event:TweenEvent){
    >> mov2 := new Tween(fondo2, "x", Regular.easeOut, fondo2.x, -550, 2, true);
    >> gotoAndStop(5);
    >> frameActual := 5;
    >>
    >> fondo.x := 550;
    >> aro.x := 550;
    >> pan2.x := 550;
    >> mov2 := new Tween(aro, "x", Regular.easeOut, aro.x, 75, 2, true);
    >> mov1 := new Tween(fondo, "x", Regular.easeOut, fondo.x, 0, 2, true);
    >> mov1 := new Tween(pan2, "x", Regular.easeOut, pan2.x, 196.05, 2, true);
    >>
    >> pan2.t1.addEventListener(MouseEvent.CLICK, T1);
    >> pan2.t2.addEventListener(MouseEvent.CLICK, T2);
    >>
    >> if(transition == 0){
    >>     transition := 1;
    >> }
    >>
    >> pan2.visible := true;
    >> mov1.addEventListener(TweenEvent.MOTION_FINISH, BtnConclusion);
    >> }

```

A continuación tenemos la función inicial de la sección de Videos. En el menú en forma de aro solo hay 4 botones que nos llevan a otros frames para ver las prácticas. En el caso de los videos se diseñó un menú desplegable desde cualquier parte de la aplicación. Se inicializa presionando el botón de la sección de Videos, el cual está ubicado en la esquina inferior izquierda.

La función GoVideos es diferente a las anteriores, ya que esta lo que hace es quitar los Listener de los botones del menú y desplegar el Sprite que se generó al principio, el cual está fuera del escenario. Una vez que acaba su movimiento invoca la función MostrarContenido.

```
//Ir al frame 6, Videos
private function GoVideos(event:MouseEvent){
    » aro.practicas.removeEventListener(MouseEvent.CLICK, Anim1);
    » aro.juegos.removeEventListener(MouseEvent.CLICK, Anim4);
    » aro.mediaa.removeEventListener(MouseEvent.CLICK, Anim3);
    » aro.tarea.removeEventListener(MouseEvent.CLICK, Anim2);
    » videos.removeEventListener(MouseEvent.CLICK, GoVideos);
    »
    » var movVideo:Tween = new Tween(extenderImagen, "x", Regular.easeOut, extenderImagen.x, -500, -2, true);
    » movVideo.addEventListener(TweenEvent.MOTION_FINISH, MostrarContenido);
}
```

Esta función tiene el propósito de crear los botones de operación dentro del Sprite. Para esto se instanciaron en variables, botónCerrar, boton1 y boton2. Hay un 4° botón pero se utiliza más adelante. El primer botón es para cerrar la sección, mientras que los otros dos son para ver el video en español o en inglés, correspondientemente.

Lo que se hace es posicionar los botones en el Sprite y luego agregarlos al mismo, además de que se agregan sus Listener correspondientes. El switch que se encuentra dentro de esta función utiliza la variable frameActual, la cual indica al programa en que frame se está actualmente, así el programa decide cual conjunto de botones van a ocultarse en base al frame en que se encuentre, esto es por mera estética.

```
//SECCION DE VIDEOS DE CONCLUSION
//Mostrar los botones de la seccion de videos, sin importar el frame
private function MostrarContenido(event:TweenEvent) {
    >> botonCerrar.x := 50;
    >> botonCerrar.y := 100;
    >> botonCerrar.scaleX := .75;
    >> botonCerrar.scaleY := .75;
    >> addChild(botonCerrar);
    >>
    >> boton1.x := 150;
    >> boton1.y := 150;
    >> addChild(boton1);
    >>
    >> boton2.x := 300;
    >> boton2.y := 150;
    >> addChild(boton2);
    >>
    >> switch(frameActual){
    >>     case 2: pan1.visible := false; break;
    >>     case 3: pan4.visible := false; break;
    >>     case 4: pan3.visible := false; break;
    >>     case 5: pan2.visible := false; break;
    >> }
    >>
    >> botonCerrar.addEventListener(MouseEvent.CLICK, CerrarVideos);
    >> boton1.addEventListener(MouseEvent.CLICK, Español);
    >> boton2.addEventListener(MouseEvent.CLICK, Ingles);
}
```

Por ultimo esta la función `CerrarVideos`, la cual se encarga de cerrar la sección de Videos. Cuando se invoca al presionar el botón cerrar, se mueve al Sprite a la posición en la que fue creado, además de quitar los botones de operación de la sección de Videos. Luego restablece los Listener de los botones del menú y des oculta el conjunto de botones pan correspondiente.

```
//Funcion que esconde la seccion de videos
private function CerrarVideos(event:MouseEvent){
    » var movVideo2:Tween = new Tween(extenderImagen, "x", Regular.easeOut, extenderImagen.x, -500, -2, true);
    » removeChild(botonCerrar);
    » removeChild(boton1);
    » removeChild(boton2);
    »
    » aro.practicas.addEventListener(MouseEvent.CLICK, Anim1);
    » aro.juegos.addEventListener(MouseEvent.CLICK, Anim4);
    » aro.mediaa.addEventListener(MouseEvent.CLICK, Anim3);
    » aro.tarea.addEventListener(MouseEvent.CLICK, Anim2);
    » videos.addEventListener(MouseEvent.CLICK, GoVideos);
    »
    » switch(frameActual){
    » » case 2: pan1.visible = true; break;
    » » case 3: pan4.visible = true; break;
    » » case 4: pan3.visible = true; break;
    » » case 5: pan2.visible = true; break;
    » }
}
}
```

Por último se tienen las funciones de los botones para ver el video en español o en inglés y para cerrarlo. Primero se explicara las funciones para ver los videos. En estas funciones se esconden los botones de operación de la sección de Videos, y se agrega un nuevo botón en la misma posición en la que se encuentra el primer botón de cerrar y se le agrega su Listener. Después creamos un nuevo Sprite, al cual le establecemos sus características y lo agregamos al escenario. Una vez hecho esto, se busca el video y se coloca en cierta posición para que quede centrado con el Sprite recientemente creado. Ya solo se agrega sobre el Sprite junto con su botón para cerrar. Lo mismo se hace para el video en Ingles.

```
var extenderImagen2:Sprite;
//VIDEOS
private function Español(event:MouseEvent){
    >> boton1.visible := false;
    >> boton2.visible := false;
    >> botonCerrar.visible := false;
    >> botonCerrar2 := new BotonCerrar();
    >> botonCerrar2.x := 50;
    >> botonCerrar2.y := 100;
    >> botonCerrar2.scaleX := .75;
    >> botonCerrar2.scaleY := .75;
    >> botonCerrar2.addEventListener(MouseEvent.CLICK, CerrarVideo1);
    >>
    >> extenderImagen2 := new Sprite();
    >> extenderImagen2.graphics.beginFill(0xFFFFFF, .30);
    >> extenderImagen2.graphics.drawRect(0, 0, 550, 400);
    >> addChild(extenderImagen2);
    >>
    >> v.playMyFlv("\\Conclusiones -español/Conclusiones -español.mp4");
    >> v.x := -600;
    >> v.y := -50;
    >> extenderImagen.addChild(v);
    >> addChild(botonCerrar2);
}
```

```

private function Ingles (event:MouseEvent) {
    >> boton1.visible := false;
    >> boton2.visible := false;
    >> botonCerrar.visible := false;
    >> botonCerrar2 := new BotonCerrar();
    >> botonCerrar2.x := 50;
    >> botonCerrar2.y := 100;
    >> botonCerrar2.scaleX := .75;
    >> botonCerrar2.scaleY := .75;
    >> botonCerrar2.addEventListener(MouseEvent.CLICK, CerrarVideo2);
    >>
    >> extenderImagen2 := new Sprite();
    >> extenderImagen2.graphics.beginFill(0xFFFFFF, .30);
    >> extenderImagen2.graphics.drawRect(0, 0, 550, 400);
    >> addChild(extenderImagen2);
    >>
    >> v.playMyFlv("\\Conclusiones ingles/Conclusiones ingles.mp4");
    >> v.x := -600;
    >> v.y := -50;
    >> extenderImagen.addChild(v);
    >> addChild(botonCerrar2);
    >>
}

```

Luego tenemos las funciones de CerrarVideo, las cuales corresponden la primera para el video en español y la segunda en inglés. Lo primero que se hace es invocar la función closeMyFlv, para cerrar el video, luego se remueven el Sprite sobre el cual se puso el video y su botón, además de des ocultar los botones de operación de la sección de Videos.

```

//FUNCION CERRAR VIDEO
//ESPAÑOL
private function CerrarVideo1(event:MouseEvent) {
    »    v.closeMyFlv("\Conclusiones ingles/Conclusiones ingles.mp4");
    »
    »    removeChild(extenderImagen2);
    »    removeChild(botonCerrar2);
    »    botonCerrar.visible := true;
    »    boton1.visible := true;
    »    boton2.visible := true;
    »
    »
}
//INGLES
private function CerrarVideo2(event:MouseEvent) {
    »    v.closeMyFlv("\Conclusiones ingles/Conclusiones ingles.mp4");
    »
    »    removeChild(extenderImagen2);
    »    removeChild(botonCerrar2);
    »    botonCerrar.visible := true;
    »    boton1.visible := true;
    »    boton2.visible := true;
    »
    »
}

```

4. CONCLUSION

La realización de esta interfaz con esa característica de que tenía que realizar una animación similar a la del giro de un cubo fue realmente un reto, en especial para encontrar la forma de simular dicha animación. Además de que se realizó un diseño que pegara con la animación y se implementó un menú desplegable, el de la sección de Videos, el cual podría usarse para el resto de menús, reduciendo el número de frames, pero aumentando las líneas de código utilizadas.