

# XEvents: Mejores Prácticas y Guía de Referencia

## Índice

[Prólogo](#)

[Guía de buenas prácticas para programar con XEvents](#)

[Descripción de la paleta básica de XEvents](#)

[Obtener coordenada X del mouse](#)

[Obtener coordenada Y del mouse](#)

[Obtener alto y ancho de la pantalla](#)

[Fijar la posición del mouse](#)

[Establecer opacidad de la línea de barrido](#)

[Establecer ancho y alto de la línea de barrido](#)

[Mostrar y ocultar línea de barrido](#)

[Establecer color del barrido](#)

[Simular click del mouse](#)

[Presionar y soltar botón del mouse](#)

[Botones del mouse precargados](#)

[Scroll del mouse](#)

[Descripción de la paleta extendida de XEvents](#)

[Simular una tecla](#)

[Simular una combinación de teclas](#)

[Teclas Precargadas](#)

[Abrir una url en el navegador](#)

[Minimizar la ventana actual](#)

[Abrir un programa instalado en el sistema](#)

[Cerrar un programa en ejecución](#)

[Detectar que un sensor fue activado](#)

[Nombrar el programa](#)

[Valor por defecto](#)

[Obtener valor de un parámetro del programa](#)

[Guardar parámetro en disco](#)

# Prólogo

Para describir esta guía, quizás sirva empezar con una aclaración sobre lo que no es:

No es un manual para empezar a programar, ni siquiera del ambiente de desarrollo TortuBots, ya que consideramos que hay un cúmulo de otros recursos que cubren ese tema.

Es una pequeña referencia para usuarios, que teniendo experiencia en su uso, ya sea en el ámbito de la introducción al pensamiento computacional o la robótica, puedan usar dicho ambiente para generar soluciones de accesibilidad.

XEvents es, en su acepción más básica y general, un plugin de TortuBots, que permite controlar e interactuar con el sistema operativo de la computadora con programas de dicho ambiente. Es por eso que en esta guía se describen las paletas del plugin y el funcionamiento de sus bloques.

Sin embargo, en su intención XEvents es mucho más que ello, porque nace con la finalidad específica de transformar a turtlebots en un ambiente para la generación de ATTs (ayudas técnicas y tecnológicas) que permitan la utilización de aplicaciones estándar por parte de usuarios con alguna discapacidad.

Para aportar en este punto, se ha sumado un aporte de buenas prácticas, que esperamos ayuden a construir ATTs fácilmente adaptables a nuevas circunstancias o adaptar las ya construidas.

Finalmente XEvents es un trabajo en curso, con muchos aspectos a mejorar, en el que es fundamental que haya gente llevándolo al límite y generando insumos para decidir en qué dirección avanzar.

# Guía de buenas prácticas para programar con XEvents

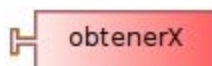
- Modularizar todo lo que sea posible el código de forma que sea más fácil de comprender no sólo para el usuario sino para los que vayan a hacer el mantenimiento del mismo.
- Contraer todas las funciones.
- Las funciones que vayan a ser ejecutadas por el usuario dejarlas arriba y con un comentario por arriba del bloque contraible, que indique que se puede hacer click sobre el bloque y que acción se va a ejecutar.
- Las funciones que no vayan a ser ejecutadas por el usuario, dejarlas hacia abajo y sin comentario para no entretener. En caso de querer poner un comentario de programación, realizarlo dentro del bloque contraible.
- Crear una acción que contenga toda la configuración posible. Para las variables, usar los box específicos para crear variables, poniendo nombres descriptivos y en español. Antes de cada box de una variable, agregar un comentario con una pequeña descripción de la misma. Esta descripción puede ser el rango de valores sobre el cual se trabaja, un valor por defecto para que el usuario tenga como referencia o cómo hacer para cambiar esa variable (por ejemplo en el caso del color).
- El programa principal, el "start", dejarlo lo más simple posible para poder entender rápidamente qué es lo que hace el programa y en lo posible a las acciones ponerles nombres en español.

## Descripción de la paleta básica de XEvents



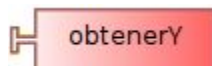
A continuación se detalla el funcionamiento de cada bloque de la paleta básica:

### Obtener coordenada X del mouse



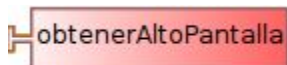
Obtiene la coordenada de la posición actual del mouse en el eje de las X.

### Obtener coordenada Y del mouse

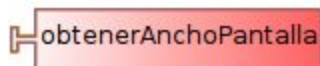


Obtiene la coordenada de la posición actual del mouse en el eje de las Y.

### Obtener alto y ancho de la pantalla



Obtiene el alto de la pantalla en pixeles.



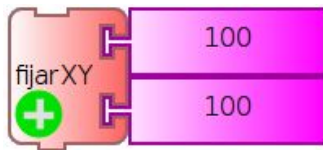
Obtiene el ancho de la pantalla en pixeles.

### Fijar la posición del mouse

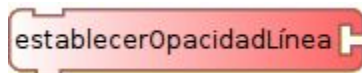


Mueve el puntero del mouse a la posición indicada.

Ejemplo de uso:

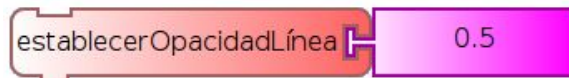


### Establecer opacidad de la línea de barrido

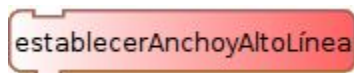


Establece la opacidad de la línea. Acepta un valor decimal entre 0 y 1, siendo 0 la opacidad mínima y 1 la opacidad total.

Ejemplo de uso:



### Establecer ancho y alto de la línea de barrido



Establece el ancho y alto de la línea de barrido.

Ejemplo de uso:



## Mostrar y ocultar línea de barrido

mostrarLínea

Muestra la línea.

ocultarLínea

Ocultar la línea.

## Establecer color del barrido

establecerColor

Establece el color de la línea de barrido.

Ejemplo de uso:

establecerColor verde = 40

## Simular click del mouse

click

Simula un click del mouse con el botón encastrado en el bloque.

Ejemplo de uso:

click clickIzquierdo

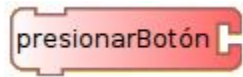
doble click

Simula un doble click del mouse con el botón encastrado en el bloque.

Ejemplo de uso:

doble click clickIzquierdo

## Presionar y soltar botón del mouse

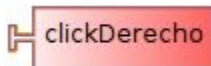


Presiona el botón encastrado en el bloque.

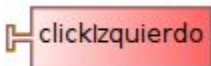


Suelta el botón encastrado en el bloque.

## Botones del mouse precargados

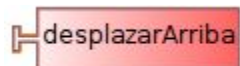


Representa el botón derecho del mouse.

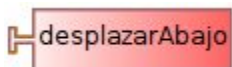


Representa el botón izquierdo del mouse.

## Scroll del mouse



Simula el botón de scroll hacia arriba.



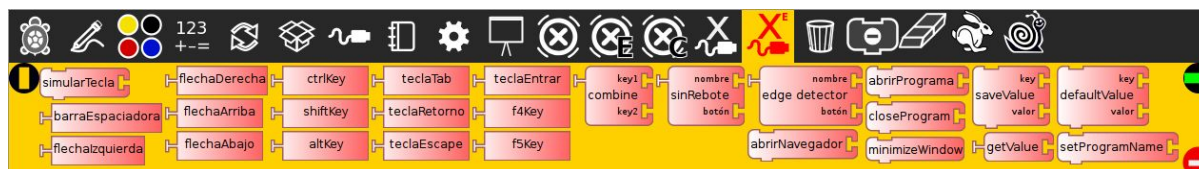
Simula el botón de scroll hacia abajo.

Ejemplo de uso:



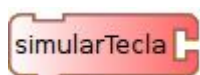
# Descripción de la paleta extendida de XEvents

La paleta extendida nos permite profundizar sobre otras funcionalidades de XEvents.



A continuación se detalla el funcionamiento de cada bloque:

## Simular una tecla



Permite simular presionar una tecla del teclado físico.

Existen 2 modos de uso:

1. Junto a un bloque de texto.



Simula presionar la tecla "a"

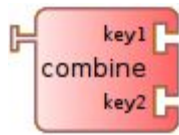
2. Junto a un bloque que represente una tecla.



Simula presionar la barra espaciadora

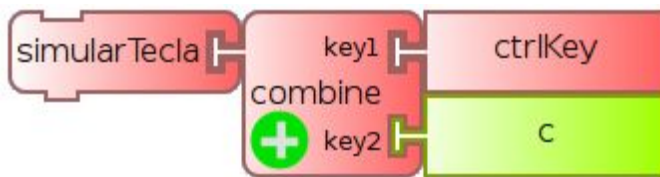


## Simular una combinación de teclas



Permite combinar dos teclas simulando presionar ambas a la vez. Primero se simula presionar la tecla encastrada en key1 (sin ser soltada) y luego se simula presionar la tecla encastrada en key2. Es útil para simular combinaciones como CTRL+C, CTRL+V etc.

Ejemplo de uso:



Simula presionar CTRL+C

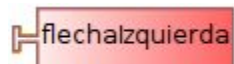
## Teclas Precargadas



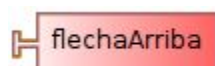
Representa la barra espaciadora. Se utiliza en conjunto con simularTecla.



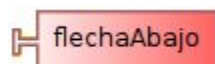
Representa la tecla flecha derecha.



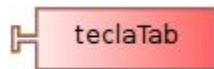
Representa la tecla flecha izquierda.



Representa la tecla flecha arriba.



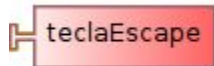
Representa la tecla flecha abajo.



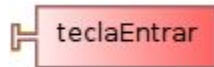
Representa la tecla tabulador.



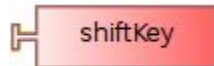
Representa la tecla de retorno.



Representa la tecla esc.



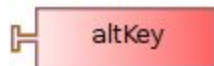
Representa la tecla enter.



Representa la tecla shift.



Representa la tecla ctrl.



Representa la tecla alt.



Representa la tecla f4.



Representa la tecla f5.

### Abrir una url en el navegador

abrirNavegador

Abre una pestaña nueva del navegador por defecto con la url que se le indique.

Ejemplo de uso:

abrirNavegador http://www.ejemplo.com

Abre una pestaña del navegador con la url [www.ejemplo.com](http://www.ejemplo.com)

### Minimizar la ventana actual

minimizeWindow

Minimiza la ventana activa en el momento de ejecución.

### Abrir un programa instalado en el sistema

abrirPrograma

Abre un programa instalado en el sistema. Se le debe indicar el nombre del binario.

Ejemplo de uso:

abrirPrograma kolourpaint

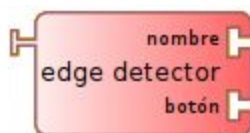
Abre el programa KolourPaint.

### Cerrar un programa en ejecución

closeProgram

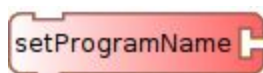
Cierra un programa abierto con el bloque abrirPrograma.

## Detectar que un sensor fue activado



El detector de flancos permite solucionar el problema de “rebotes” en las señales leídas de los sensores.

## Nombrar el programa



Le asigna un nombre al programa. Sirve para poder usar parámetros con el mismo nombre en diferentes programas sin que haya interferencia.

Define el espacio de nombres de los parámetros manipulados por los bloques: Valor por defecto, leer valor y grabar valor.

Ejemplo de uso:



## Valor por defecto



Este bloque define un parámetro del programa actual, asignándole un valor por defecto en caso que no haya una parametrización previa para el usuario.

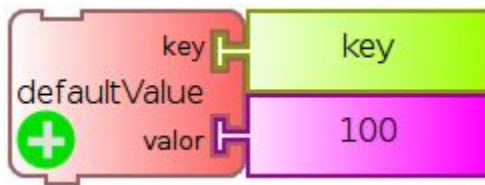
Genera una variable homónima, haciendo accesible el valor del parámetro, al resto del programa.

Se diferencia de las variables en que su valor inicial puede mantenerse o ser modificado en forma independiente del programa que lo define.

La utilización de estos parámetros permite adaptar un programa sin tener que modificarlo, algo útil especialmente si dicho programa se exportó como código python para ser utilizado independientemente de turtlebots.

Es recomendable que este bloque se encuentre en todos los programas que utilizan un cierto parámetro, utilizándose el primer valor asignado.

Ejemplo de uso:



### Obtener valor de un parámetro del programa



Obtiene el valor, previamente salvado, del parámetro del programa. En caso que no haya uno salvado, utiliza el valor por defecto. El valor obtenido no es influido por el contenido de la variable homónima al parámetro.

Ejemplo de uso:



Obtiene el valor guardado del parámetro "key".

### Guardar parámetro en disco



Asigna valor a un parámetro, y lo guarda en el sistema.

En general este comando se utiliza para establecer la parametrización, en un programa distinto al que utiliza el bloque "valor por defecto". Sin embargo se pueden utilizar ambos bloques en conjunto, para mantener valores actualizados, por ejemplo en el caso de guardar un cambio de velocidad de lectura o barrido.

Ejemplo de uso:

