

SERVIDOR GUI

Contenido

Descripción.....	3
Sintaxis de comandos.....	3
ID's de comandos:	3
Comando setTitle (String)	3
Comando setSize (x, y)	3
Comando setLocation (x, y).....	3
Comando setVisible (true/false)	3
Comando setResizable (true/false).....	3
Comando drawPixel (x, y, Color (r, g, b)).....	4
Comando updateScreen.....	4
Comando autoUpdateScreen (true/false).....	4
Comando getKeyboard.....	4

Descripción

Los comandos serán enviados desde un cliente cualquier. El servidor ejecutara una GUI el cual se vera afectada por los comandos externos de los clientes.

Sintaxis de comandos

Los comandos se definen a través de **ID1**<parametro1, parametro2, ...>: **ID2**<parametro1, parametro2, ...>: ...

Los argumentos compuestos como aquellos que representan datos de objetos, y cuentan como un único parámetro. Los argumentos compuestos se forman por paréntesis, por ejemplo, un objeto “Colors” con 3 valores (r, g, b) se vería de la siguiente manera:

drawPixel<350, 350, (255,255,255)>

Esto pintaría un pixel en las coordenadas (350,350) de color blanco (255,255,255).

Los comandos pueden devolver información. La sintaxis es la siguiente:

return[id]<valor1, valor2, ...>

Ejemplo, el comando **getKeyboard**. Comando ejecutado:

getKeyboard<>

retorna las teclas pulsadas:

return[getKeyboard]<“a”, “b”, “c”>

ID's de comandos:

Comando setTitle (String)

Permite al cliente cambiar el titulo de la ventana ser servidor.

setTitle<“Titulo de ventana”>

Comando setSize (x, y)

Permite al cliente cambiar el tamaño de la ventana del servidor.

setSize<100, 200>

Comando setLocation (x, y)

Perite al cliente cambiar la localización de la ventana del servidor.

setLocation<100, 200>

Comando setVisible (true/false)

Permite hacer visible o no la ventana en el servidor.

setVisible<true>

Comando setResizable (true/false)

Permite indicar si la ventana se puede redimensionar o no.

Server GUI

`setResizable<true>`

Comando `drawPixel (x, y, Color (r, g, b))`

Permite dibujar un Pixel en unas coordenadas dadas con un color específico en RGB.

`drawPixel<100, 200, (255,255,0) >`

Comando `updateScreen`

Permite actualizar la pantalla una vez

`updateScreen<>`

Comando `autoUpdateScreen (true/false)`

Permite que la GUI se auto actualicé. El comando `updateScreen` no tendrá efecto sobre este modo.

`autoUpdateScreen<true>`

Comando `getKeyboard`

Obtiene las teclas pulsadas.

`getKeyboard<>`

retorno: `return[getKeyboard]<"a", "b", "c">`