# ***Curso basico de MATLAB***

## 27. Interfaz grafica de usuario: GUI

Una interfaz grafica es el vinculo entre el usuario y un programa computacional, constituida generalmente por un conjunto de comandos o menúes, instrumentos y métodos por medio de los cuales el usuario se comunica con el programa durante las operaciones que se desean realizar, facilitando la entrada y salida de datos e información.
Aunque un programa sea muy poderoso, si se manipula por medio de una interfaz grafica pobremente elaborada, tendrá poco valor para un usuario inexperto. Es por ello que las interfaces graficas tienen una gran importancia para usuarios inexpertos o avanzados de cualquier programa ya que facilita su uso.
***GUI*** es un entorno de programación visual disponible en ***Matlab*** para realizar y ejecutar programas que necesiten ingreso continuo de datos.
Tiene las características básicas de todos los programas visuales como ***Visual Basic*** o ***Visual C++.***
Cuando creamos una ***GUI***, se generan dos archivos: uno con extensión ***.m*** y otro con extensión ***.fig***

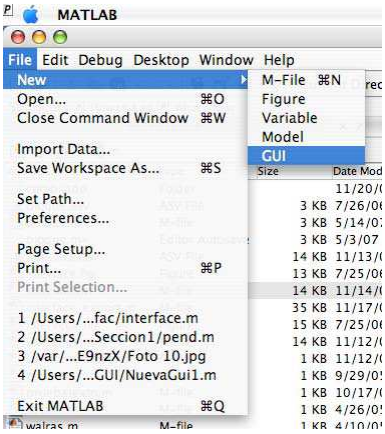
### *27.1 Crear una GUI*

Para crear una GUI, lo podemos hacer de tres maneras:

- ***Ejecutando la siguiente instrucción en la ventana de comandos:***

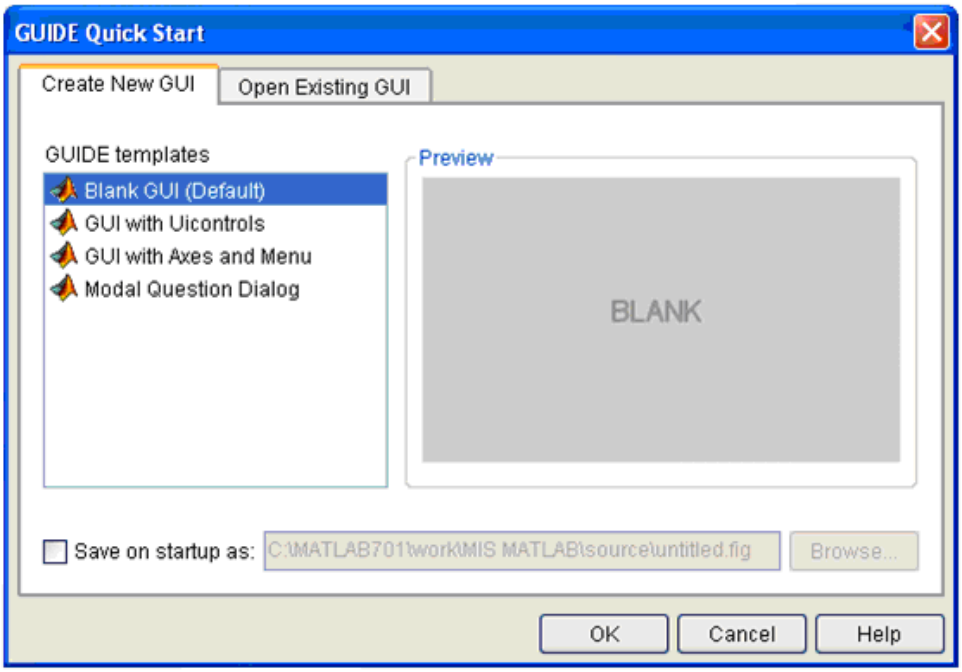
  >> guide

- ***Vamos a File-> New->GUI :***




- ***Haciendo un clic en el ícono***

Aparecerá la siguiente pantalla




Se presentan las siguientes opciones:

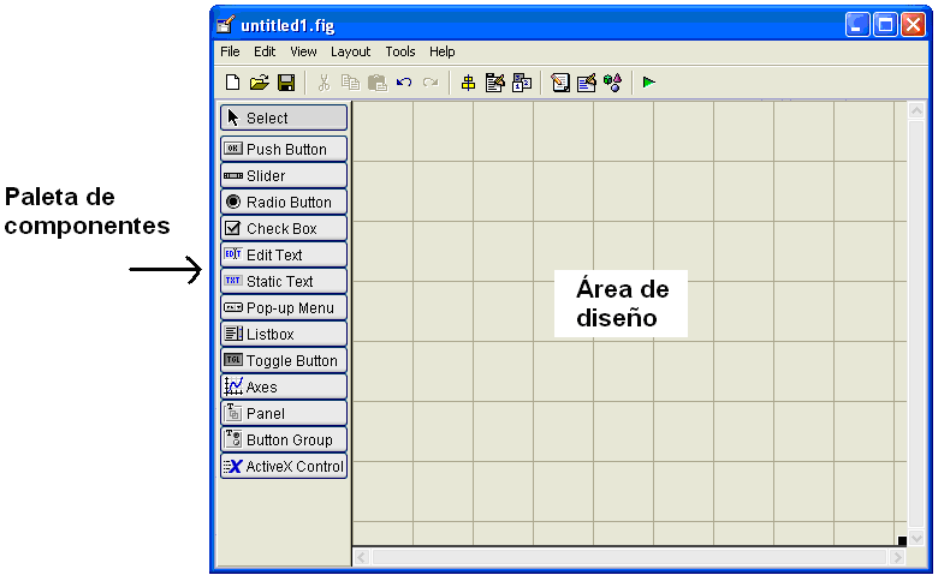
- **Blank GUI (Default):** La opción de interfaz gráfica de usuario en blanco (viene predeterminada), nos presenta un formulario nuevo, en el cual podemos diseñar nuestro programa.

- **GUI with Uicontrols:** Esta opción presenta un ejemplo en el cual se calcula la masa, dada la densidad y el volumen, en alguno de los dos sistemas de unidades. Podemos ejecutar este ejemplo y obtener resultados.

- **GUI with Axes and Menu:** Esta opción es otro ejemplo el cual contiene el menú File con las opciones Open, Print y Close. En el formulario tiene un *Popup menu*, un *push button* y un objeto *Axes*, podemos ejecutar el programa eligiendo alguna de las seis opciones que se encuentran en el menú despegable y haciendo clic en el botón de comando.

- **Modal Question Dialog:** Con esta opción se muestra en la pantalla un cuadro de diálogo común, el cual consta de una pequeña imagen, una etiqueta y dos botones *Yes* y *No*, dependiendo del botón que se presione, el GUI retorna el texto seleccionado (la cadena de caracteres ‘Yes’ o ‘No’).

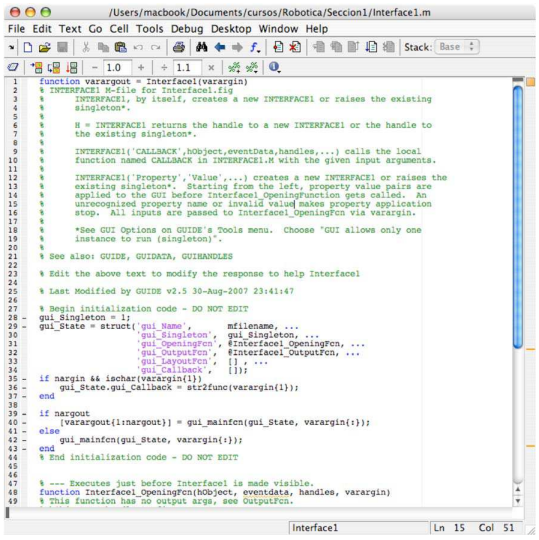
Elegimos la primera opción, ***Blank GUI***, y tenemos:




Asimismo, tenemos también las siguientes herramientas:

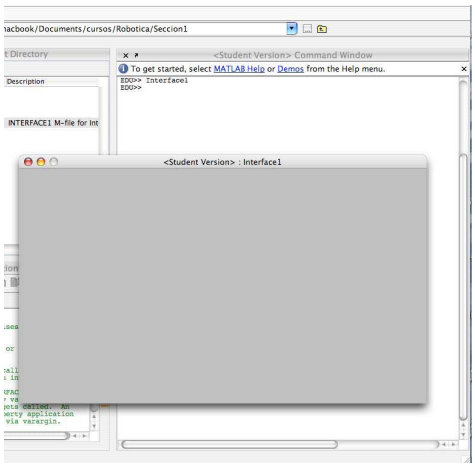
| | |
|---|---|
| | Alinear objetos |
| | Editor de menú |
| | Editor de orden de etiqueta |
| | Editor del M-file |
| | Propiedades |
| | Navegador |
| | Grabar y ejecutar |

Como mencionamos anteriormente, se generaran dos archivos: uno con el nombre **untitled.m** y otro con el nombre **untitled.fig.** Luego de renombrarlos con el nombre **interface1** es conveniente explorarlos para ver las diferencias que se van a ir produciendo a medida que se vayan introduciendo elementos dentro de la interfaz.



Lo que vemos aquí es el archivo de tipo ***function*** con la extensión característica de ***Matlab* .m**. Este archivo es construido automáticamente por ***Matlab*** y las líneas de código que aparecen son las que crean la interfaz que aparece en el archivo **interface1.fig**.
El archivo de inicialización define los parámetros básicos de la interfaz y crea un conjunto de ***handles (identificador de cada componente)*** para cada uno de los objetos que vayan apareciendo sobre la interfaz.

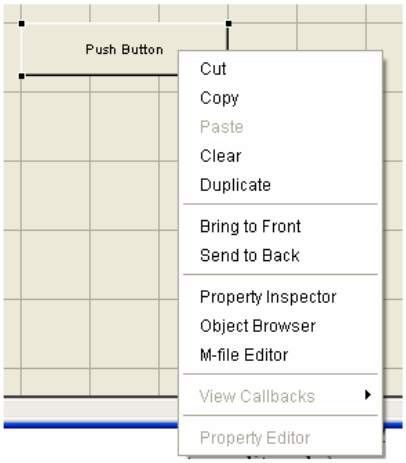
Si hacemos clic sobre el archivo **interface1.fig**, o invocamos a **interface1.m** en la ventana de comandos obtenemos:



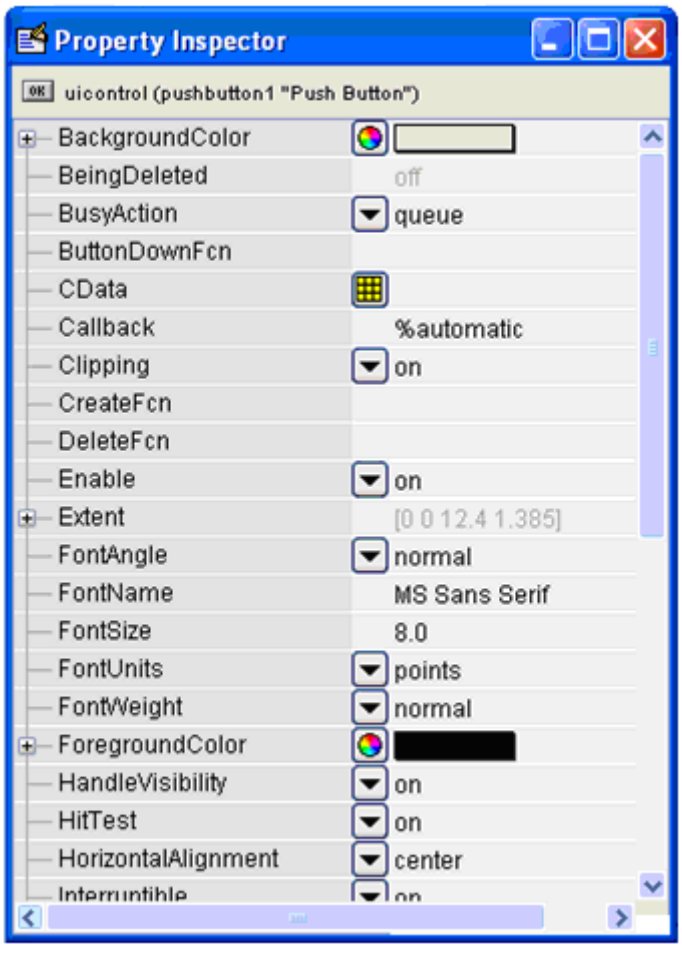

Como se puede apreciar, la interfaz no contiene nada más que un lienzo vacío sin ninguna funcionalidad. El código de inicialización que está en este momento escrito en **Interface1.m** se encarga de crear la interfaz tal y como está, y por esa razón ese código no debe ser modificado.

## *27.2 Propiedades de los componentes*

Si insertamos algún elemento de la paleta de componentes en el área de diseño y hacemos clic derecho sobre dicho objeto, obtenemos una ventana desde la que podemos cambiar las propiedades del objeto que hemos agregado:




La opción ***Property Inspector*** nos permite personalizar cada elemento.



## *27.3 Creación de subrutinas para los componentes*

Al hacer clic derecho en el elemento ubicado en el área de diseño, una de las opciones más importantes es ***View Callbacks***, la cual, al ejecutarla, abre el archivo ***.m*** asociado a nuestro diseño y nos posiciona en la parte del programa que corresponde a la subrutina que se ejecutará cuando se realice una determinada acción sobre el elemento que estamos editando.
Por ejemplo, al ejecutar ***View Callbacks*** en el ***Push Button***, nos ubicaremos en la parte del programa:

```
function pushbutton1_Callback(hObject, eventdata, handles)
%hObject handle to pushbutton1 (see GCBO)
%eventdata reserved-to be defined in a future version of MATLAB
%handles structure with handles and user data (see GUIDATA)
```

## *27.4 Sentencias GET y SET*

La asignación u obtención de valores de los componentes se realiza mediante las sentencias ***get*** y ***set.***

Para que la variable *up_down* tenga el valor del ***slider*** etiquetado como *slider1* escribimos:

```
up_down= get(handles.slider1,'Value');
```

*Notar que siempre se obtienen los datos a través del puntero handles.*

Para asignar el valor a la variable *up_down* al *statictext* etiquetado como ***text1*** escribimos:

```
set(handles.text1,'String',up_down);
```

## *27.5 Ejemplo de una GUI compleja*