

# Praktikum MATLAB – Modul 8

## GUI

### Tujuan Praktikum

- Mahasiswa dapat membuat grafik user interface (GUI) pada MATLAB.
- Mahasiswa mampu membuat aplikasi GUI sederhana pada MATLAB.

### Alat dan Bahan

Untuk praktikum ini, diperlukan alat dan bahan sebagai berikut:

*Tabel 2.1 Alat dan bahan*

No	Nama	Banyak	Keterangan
1	Komputer / Laptop	1	Disiapkan sendiri
2	Matlab	1	Disiapkan sendiri

### Pendahuluan

GUIDE atau GUI builder merupakan sebuah graphical user interface (GUI) yang dibangun dengan obyek grafik seperti tombol (button), kotak teks, slider, menu dan lain-lain. Aplikasi yang menggunakan GUI umumnya lebih mudah dipelajari dan digunakan karena orang yang menjalankannya tidak perlu mengetahui perintah yang ada dan bagaimana kerjanya.

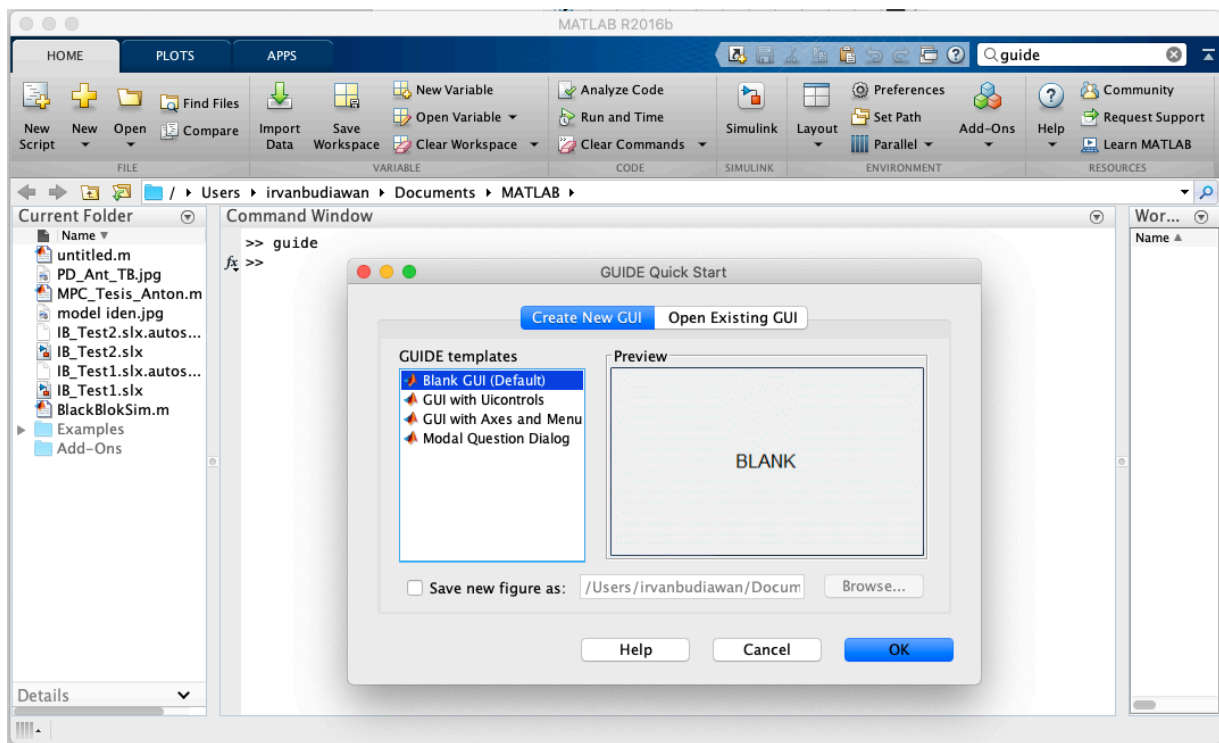
Sampai saat ini, jika kita membicarakan pemrograman berorientasi visual, yang ada di benak kita adalah sederetan bahasa pemrograman, seperti visual basic, Delphi, visual C++, visual Fox Pro, dan lainnya yang memang didesain secara khusus untuk itu. Matlab merintis ke arah pemrograman yang menggunakan GUI dimulai dari versi 5.

GUIDE Matlab mempunyai kelebihan tersendiri dibandingkan dengan bahasa pemrogram lainnya, diantaranya:

1. GUIDE Matlab banyak digunakan dan cocok untuk aplikasi-aplikasi berorientasi sains, sehingga banyak peneliti dan mahasiswa menggunakan GUIDE Matlab untuk menyelesaikan riset atau tugas akhirnya.
2. GUIDE Matlab mempunyai fungsi built-in yang siap digunakan dan pemakai tidak perlu repot membuatnya sendiri.
3. Ukuran file, baik FIG-file maupun M-file, yang dihasilkan relatif kecil.
4. Kemampuan grafisnya cukup andal dan tidak kalah dibandingkan dengan bahasa pemrograman lainnya.

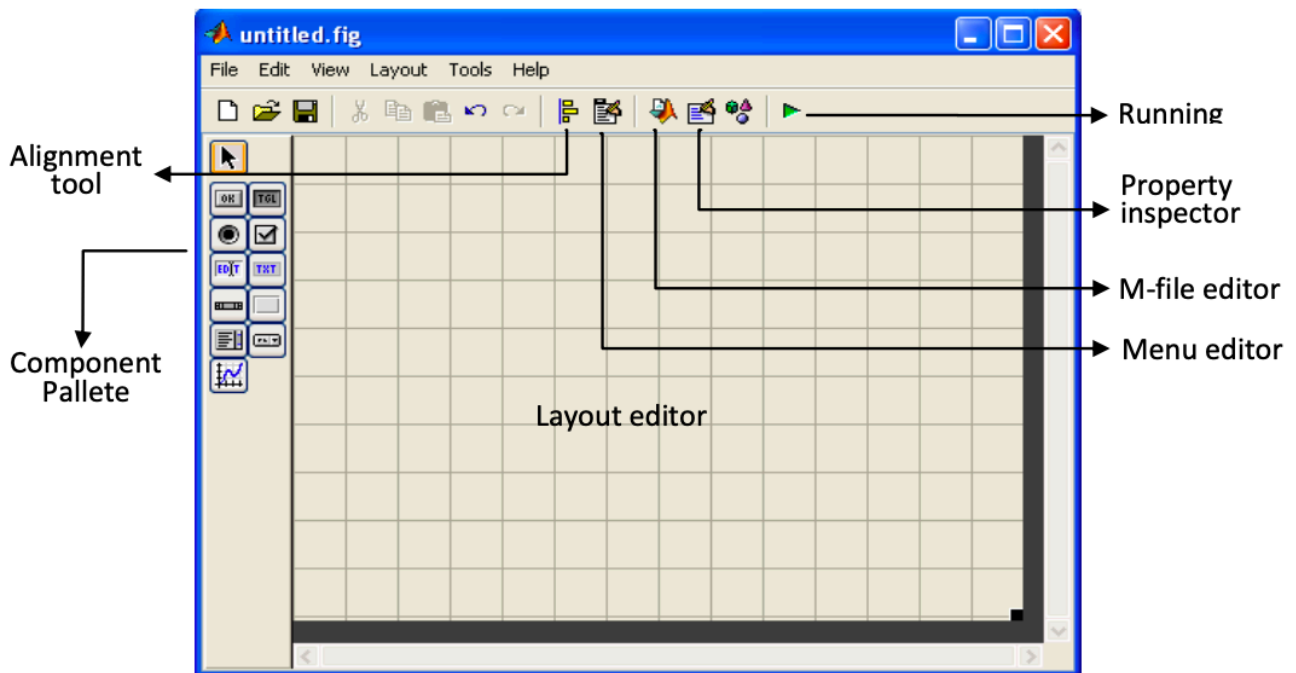
# GUI MATLAB

Memulai GUIDE/GUI Matlab, pada Command matlab ketikan: `>> guide`, akan muncul Gambar 1

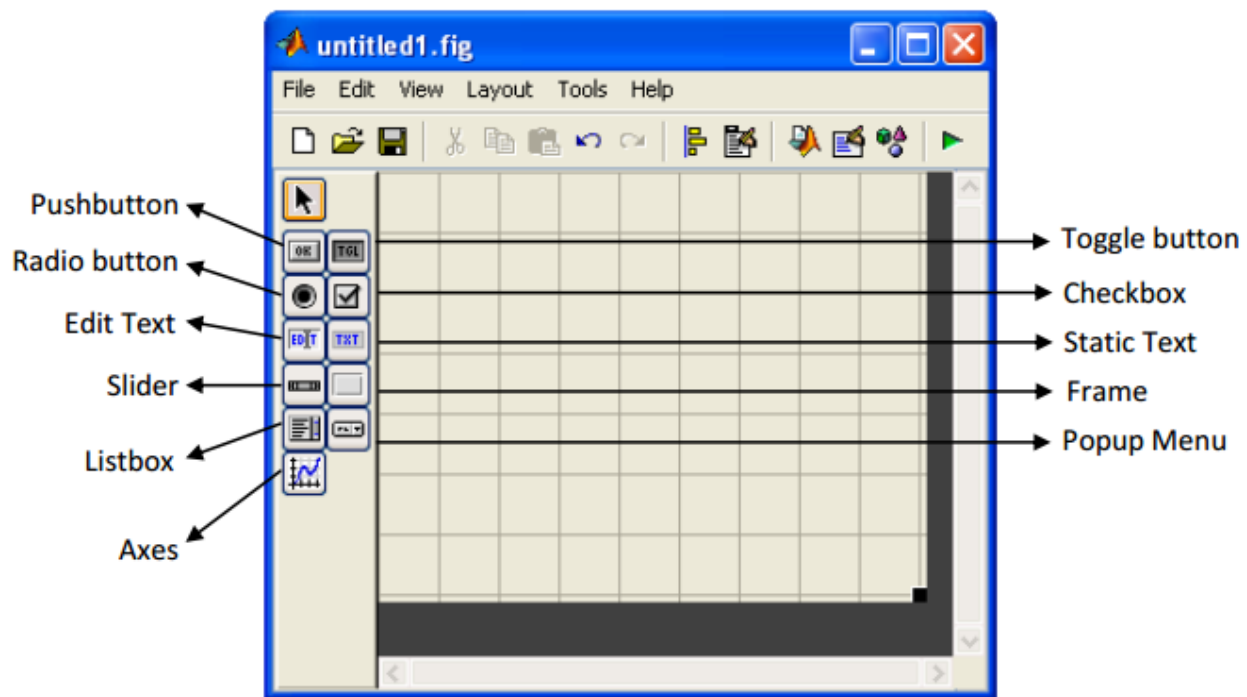


*Gambar 1 Tampilan awal GUIDE*

Pilih Blank GUI (Default) dan klik OK untuk memulai membuat gui. Tampilan kerja untuk membuat gui diperlihatkan pada Gambar 2.



*Gambar 2 Area kerja gui pada MATLAB*

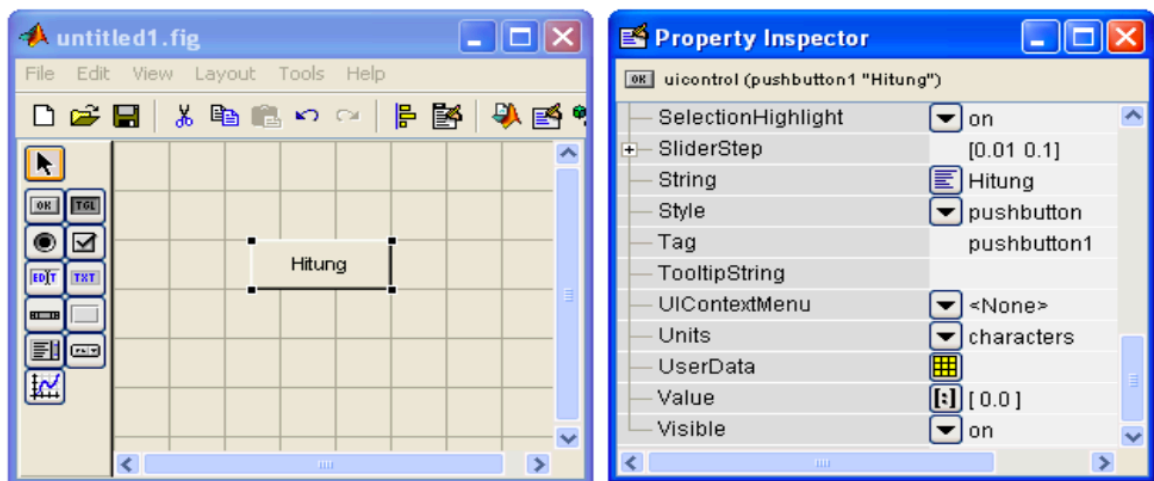


*Gambar 3 Komponen palet*

Komponen palet pada GUIDE Matlab terdiri dari beberapa uicontrol (kontrol user interface), seperti pada bahasa pemrograman visual lainnya, yaitu: pushbutton, togglebutton, radiobutton, checkboxes, edit text, static text, slider, frames, listboxes, popup menu, dan axes. Kita dapat meletakkan semua kontrol pada layout editor dan selanjutnya hanya tinggal mengaturnya melalui property inspector.

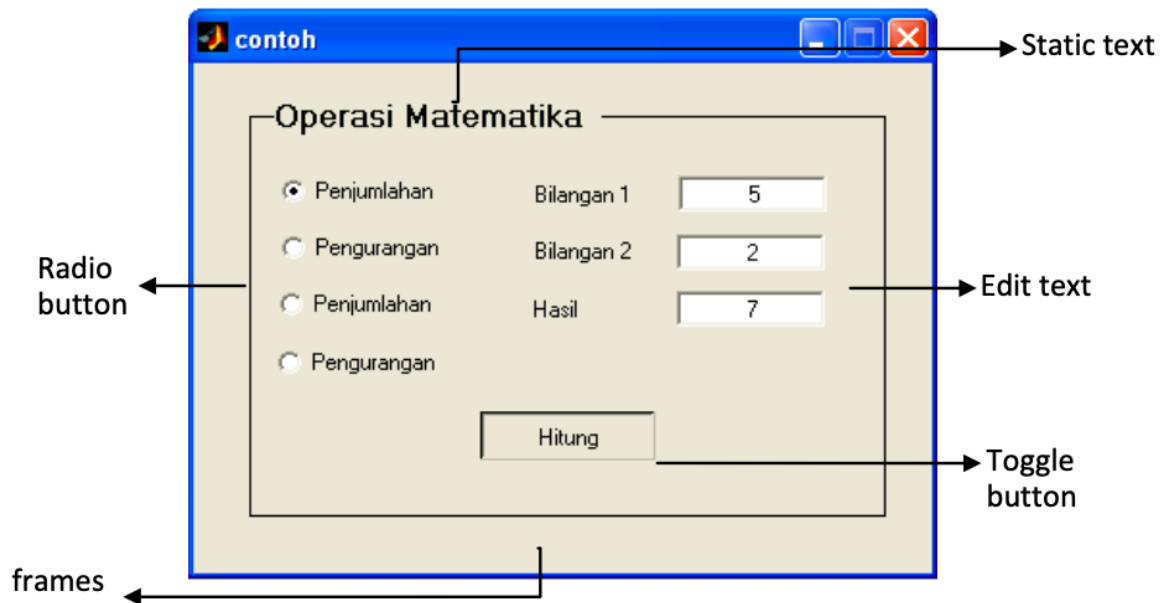
Semua kontrol pada GUIDE dapat dimunculkan pada layout/figure dengan cara mendrag kiri kontrol yang diinginkan ke figure. Adapun penjelasan fungsi masing-masing kontrol adalah sebagai berikut:

- a) Pushbutton  
Pushbutton merupakan jenis kontrol berupa tombol tekan yang akan menghasilkan tindakan jika diklik, misalnya tombol OK, Cancel, Hitung, Hapus, dan sebagainya. Untuk menampilkan tulisan pada pushbutton kita dapat mengaturnya melalui property inspector dengan mengklik objek pushbutton pada figure, lalu mengklik toolbar property inspector atau menggunakan klik kanan lalu pilih property inspector. Selanjutnya isilah tab string dengan label yang diinginkan, misalnya Hitung.



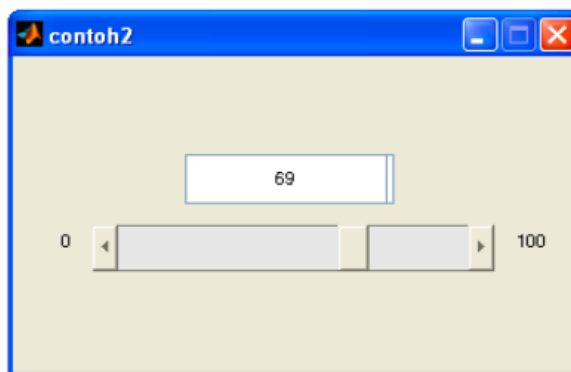
*Gambar 4 Komponen palet pushbutton*

- b) Toggle Button  
Toggle button memiliki fungsi yang sama dengan pushbutton. Perbedaannya adalah saat pushbutton ditekan, maka tombol akan kembali pada posisi semula jika tombol mouse dilepas, sedangkan pada toggle button, tombol tidak akan kembali ke posisi semula, kecuali kita menekannya kembali. Contoh aplikasi Radio button dapat dilihat pada Gambar 5.
- c) Radio Button  
Radio button digunakan untuk memilih atau menandai satu pilihan dari beberapa pilihan yang ada. Misalnya, kita membuat aplikasi operasi Matematika (penjumlahan, pengurangan, perkalian, dan pembagian). Contoh aplikasi Radio button dapat dilihat pada Gambar 5.
- d) Edit Text dan Static Text  
Edit text digunakan untuk memasukkan atau memodifikasi suatu text yang diinputkan dari keyboard, sedangkan static text hanya berguna untuk menampilkan text/tulisan, sehingga kita tidak bisa memodifikasi/mengedit text tersebut kecuali melalui property inspector. Contoh edit dan static text dapat dilihat pada Gambar 5.
- e) Frames  
Frames merupakan kotak tertutup yang dapat digunakan untuk mengelompokkan kontrol-kontrol yang berhubungan. Tidak seperti kontrol lainnya, frames tidak memiliki rutin callback.



*Gambar 5 Penggunaan toggle button, radio button, edit text, static text dan frame*

- f) Checkboxes  
Kontrol checkboxes berguna jika kita menyediakan beberapa pilihan mandiri atau tidak bergantung dengan pilihan-pilihan lainnya. Contoh aplikasi penggunaan checkboxes adalah ketika kita diminta untuk memilih hobi. Karena hobi bisa lebih dari satu, maka kita dapat mengklik checkboxes lebih dari satu kali.
- g) Slider  
Slider berguna jika kita menginginkan inputan nilai tidak menggunakan keyboard, tetapi hanya dengan cara menggeser slider secara vertical maupun horizontal ke nilai yang kita inginkan. Dengan menggunakan slider, kita lebih fleksibel dalam melakukan pemasukan nilai data karena kita dapat mengatur sendiri nilai max, nilai min, serta sliderstep.



*Gambar 6 Komponen slider*

- h) Popup Menu  
Popup menu berguna menampilkan daftar pilihan yang didefinisikan pada String Property ketika mengklik tanda panah pada aplikasi dan memiliki fungsi yang sama seperti radio button. Ketika tidak dibuka, popup menu hanya menampilkan satu item yang menjadi pilihan pertama pada String Property. Popup menu sangat bermanfaat ketika kita ingin memberi sebuah pilihan tanpa jarak, tidak seperti radiobutton.
- i) Axes

Axes berguna untuk menampilkan sebuah grafik atau gambar (image). Axes sebenarnya tidak masuk dalam UIControl, tetapi axes dapat diprogram agar pemakai dapat berinteraksi dengan axes dan obyek grafik yang ditampilkan melalui axes.

### **Membuat GUI dengan MATLAB**

MATLAB mengimplementasikan GUI sebagai sebuah figure yang berisi berbagai style obyek UIControl. Selanjutnya kita harus memprogram masing-masing obyek agar dapat bekerja ketika diaktifkan oleh pemakai GUI.

Langkah dasar yang harus dikerjakan dalam membuat GUI adalah:

#### **1) Mengatur layout komponen GUI**

Setelah kita membuka GUIDE Matlab dan telah menentukan template GUI, langkah selanjutnya adalah mendesain figure dengan menggunakan komponen palet sesuai dengan kebutuhan, seperti pushbutton, radiobutton, checkboxes, edit text, static text, slider, frames, popup menu, axes, dan sebagainya. Selanjutnya kita dapat mengatur layout masing-masing komponen, baik string(caption), font, color, size, dan sebagainya menggunakan property inspector. Jika kita telah selesai mendesain, jangan lupa untuk menyimpan file figure yang secara default akan memiliki ekstensi \*.fig. Dari sini, matlab secara otomatis akan membuatkan sebuah m-file dengan nama yang sama, yaitu file berekstensi \*.m.

#### **2) Memprogram Komponen GUI**

M-file yang telah dibuat pada langkah sebelumnya, akan otomatis terbuka dan kita harus menulis programnya agar komponen kontrol dapat bekerja secara simultan. Untuk membuat program dalam m-file kita cukup memperhatikan fungsi-fungsi matlab bertanda callback dimana perintah disisipkan.

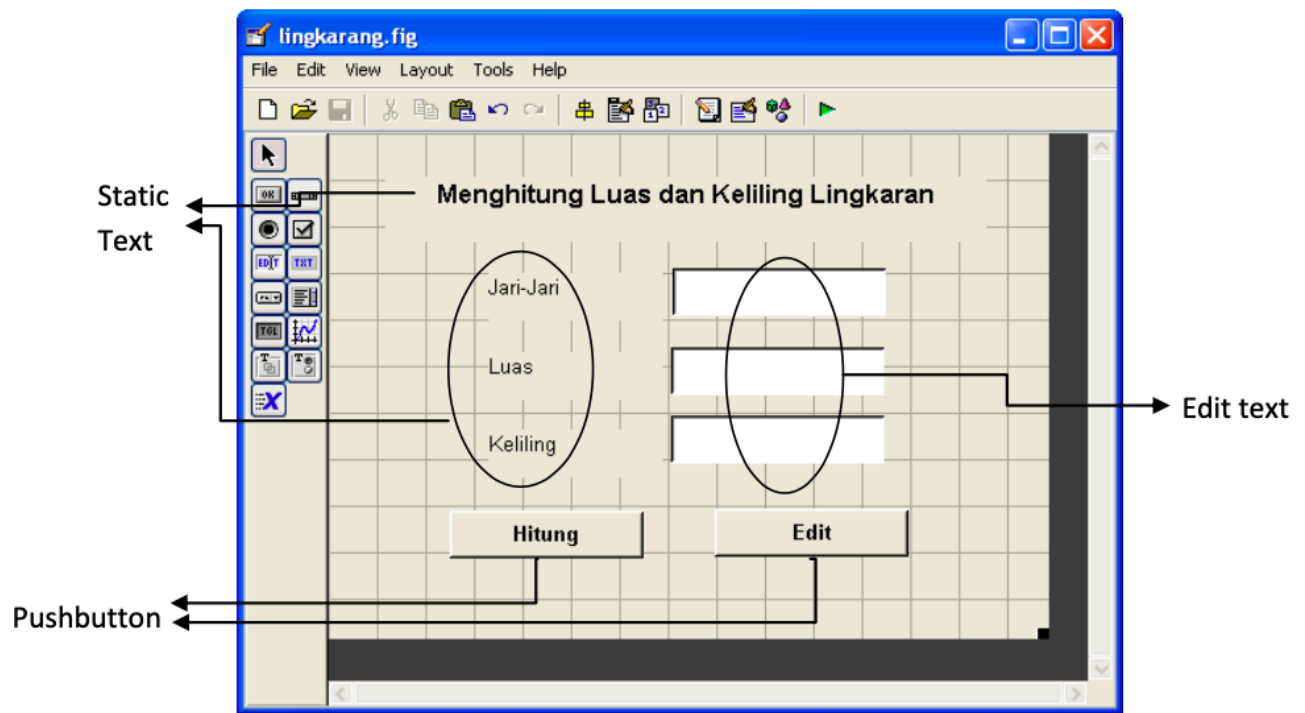
Dari langkah-langkah dasar diatas, secara sederhana sebenarnya GUI Matlab dibentuk oleh dua buah file, yaitu fig-file dan m-file.

Contoh membuat aplikasi untuk menghitung Luas dan Keliling Lingkaran, langkah-langkahnya sebagai berikut:

##### **a) Mendesain Figure**

Dalam mendesain figure, kita harus dapat membayangkan komponen apa saja yang perlu kita tampilkan. Seperti dalam membuat aplikasi menghitung luas dan keliling lingkaran, variabel input yang dibutuhkan adalah jari-jari, kemudian variabel outputnya adalah luas dan keliling lingkaran, maka kita memerlukan 3 edit text yang digunakan menampilkan variabel-variabel tersebut. Kemudian kita juga memerlukan 1 tombol pushbutton/togglebutton untuk mulai melakukan proses perhitungan. Kita juga bisa menambahkan komponen lain untuk memperjelas dan mempercantik desain figure yang akan kita buat.

Desainlah figure seperti pada Gambar 7. Gunakan 4 buah static text, 3 buah edit text, dan 2 buah pushbutton. Dalam meletakkan komponen palette boleh tidak sesuai dengan Gambar 7.



*Gambar 7 GUI menghitung luas dan keliling lingkaran*

b) Mengatur Layout Komponen

Setelah kita selesai mendesain figure, aturlah masing-masing komponen menggunakan property inspector.

*Tabel 1 Property Inspector GUI menghitung luas dan keliling lingkaran*

Komponen	Property Inspector			
	FontSize	FontWeight	String	Tag
Static text 1	12	`Bold	Menghitung Luas dan Keliling Lingkaran	text1
Static text 2	10	Normal (default)	Jar-Jari	text2
Static text 3	10	Normal (default)	Luas	text3
Static text 4	10	Normal (default)	Keliling	text4
Edit text 1	10	Normal (default)	Kosongkan	edit1
Edit text 2	10	Normal (default)	Kosongkan	edit2
Edit text 3	10	Normal (default)	Kosongkan	edit3
Pushbutton 1	10	`Bold	Hitung	btn_hitung
Pushbutton 2	10	`Bold	Edit	btn_exit

c) Menyimpan figure

Setelah selesai mendesain

lingkarang.fig, secara otomatis kita akan dibuatkan kerangka m-file dengan nama yang sama, yaitu lingkaran.m.

Dari beberapa fungsi yang muncul di m-file. Kita cukup memperhatikan fungsi yang memiliki callback,

yaitu edit1\_Callback, edit2\_Callback, edit3\_Callback, btn\_hitung\_Callback, dan btn\_edit\_Callback.

- edit1

Di bawah function edit1\_callback, tambahkan program menjadi berikut:

```
jari=str2num(get(handles.edit1, 'String'));  
handles.jari=jari;  
guidata(hObject, handles)
```

- edit2 dan edit3

untuk function edit1\_callback dan function edit2\_callback, kita tidak perlu menambahkan kode apapun dibawahnya, karena hanya berfungsi untuk menampilkan hasil dari perhitungan.

- btn\_hitung

Di bawah function btn\_hitung\_callback, tambahkan program menjadi berikut:

```
%kode ini dieksekusi jika kita menekan tombol hitung  
jari=handles.jari;  
luas=pi*jari*jari;  
kel=2*pi*jari;  
  
%menampilkan hasil perhitungan  
set(handles.edit2, 'string', luas);  
set(handles.edit3, 'string', kel);
```

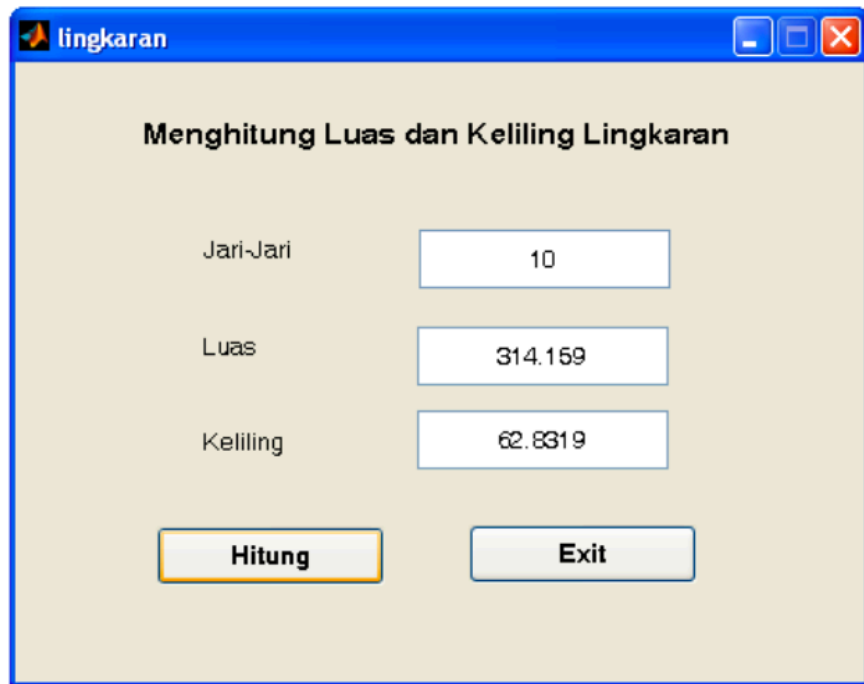
- btn\_ext

```
%untuk keluar dari aplikasi  
delete(handles.figure1)
```

#### d) Running GU

Setelah langkah-langkah diatas dijalankan, langkah terakhir adalah menjalankan aplikasi yang telah dibuat dengan mengklik tombol Run dari jendela figure atau dari jendela debug m-file (tekan F5), sehingga akan muncul tampilan berikut.



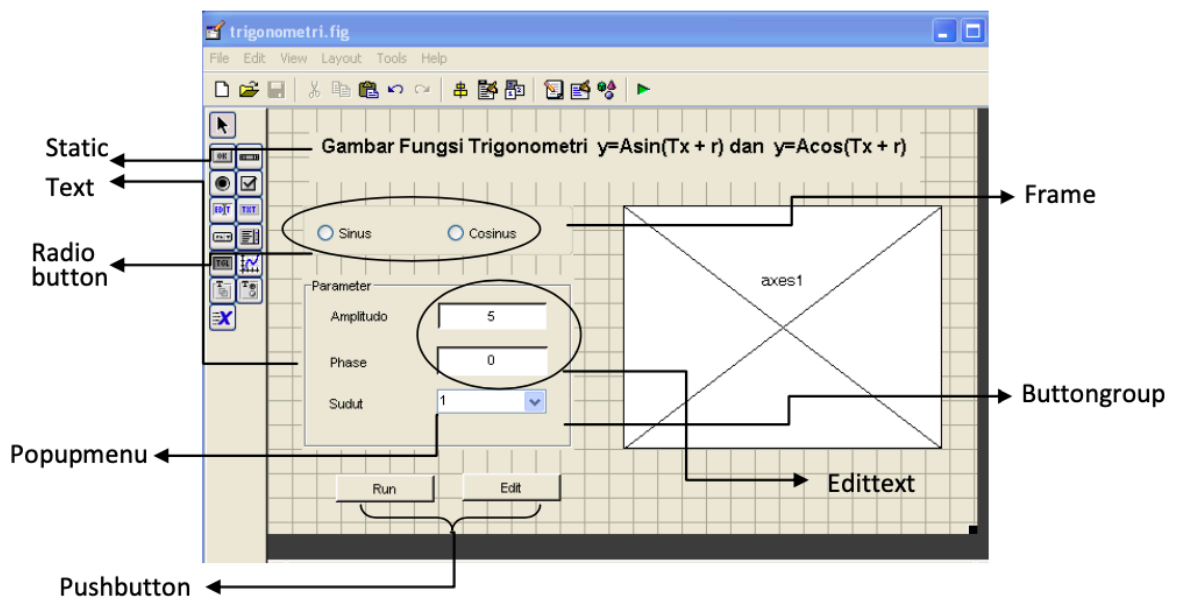


*Gambar 8 Tampilan akhir aplikasi menghitung luas dan keliling lingkaran*

contoh 2:

a) Mendesain Figure

Desain figure seperti dalam Gambar 9. Dalam meletakkan komponen palette boleh tidak sesuai dengan gambar (sesuai selera).



*Gambar 9 GUI Fungsi trigonometri*

b) Mengatur Layout Komponen

Setelah kita selesai mendesain figure, aturlah masing-masing komponen menggunakan property inspector.

*Tabel 2 Property Inspector GUI fungsi trigonometri*

Komponen	Property Inspector			
	FontSize	FontWeight	String/Title	Tag
Static text 1	12	`Bold	Gambar Trigonometri $y = \text{Asin}(Tx+r)$ dan $y = \text{Acos}(Tx+r)$	text1
Static text 2	Tetap (default 8)	Tetap (default Normal)	Amplitudo	text2
Static text 3	tetap	tetap	Phase	text3
Static text 4	tetap	tetap)	Sudut	text4
Edit text 1	tetap	tetap	5	edit1
Edit text 2	tetap	tetap	0	edit2
Popupmenu	tetap	tetap	1, 2, 3, 4	popupmenu
Pushbutton 1	tetap	`Bold	Run	pushbutton1
Pushbutton 2	tetap	`Bold	Edit	pushbutton2
Radiobutton 1	tetap	tetap	Sinus	radiobutton1
Radiobutton 1	tetap	tetap	Cosinus	radiobutton2
Button Group	tetap	tetap	Parameter	

c) Menyimpan Figure

Setelah selesai mendesain figure, langkah selanjutnya adalah menyimpan figure, beri nama trigonometri.fig, secara otomatis kita akan dibuatkan kerangka m-file dengan nama yang sama, yaitu trigonometri.m.

Dari beberapa fungsi yang muncul di m-file. Kita cukup memperhatikan fungsi yang memiliki callback, yaitu edit1\_Callback, edit2\_Callback, edit3\_Callback, btn\_hitung\_Callback, dan btn\_edit\_Callback.

- Popupmenu1

Di bawah function popupmenu1\_callback, tambahkan program menjadi berikut:

```
indeks=get(handles.popupmenu1,'Value');
handles.indeks=indeks;
guidata(hObject, handles)
```

- Radiobutton1

```
set(handles.radiobutton2,'Value',0);
pilih = 1; handles.pilih=pilih;
guidata(hObject,handles)
```

- Radiobutton2

```
set(handles.radiobutton1, 'Value', 0);

pilih = 2; handles.pilih=pilih;

guidata(hObject,handles)
```

- Pushbutton2

```
set(handles.edit1, 'string', ''); %mengosongkan
string pada edit1

set(handles.edit2, 'string', ''); %mengosongkan
string pada edit2

axes(handles.axes1);

plot(0); %membersihkan gambar pada axes
```

- Pushbutton1

```
indeks=get(handles.popupmenu2, 'Value'); %mengambil
nilai

if indeks == 1 %jika menekan pilihan 1

    T=1; %Maka T=1;

    else if indeks==2 %jika menekan pilihan 2

        T=2 %maka T=2, dst

    else if indeks==3

        T=3;

    else if indeks==4

        T=4;

    end

end

end

end

end
```

```

A=str2num(get(handles.edit1, 'string')); %mengambil
nilai pada edit1

r=str2num(get(handles.edit2, 'string')); %mengambil
nilai pada edit2

x=-10:0.1:10; %membangkitkan sumbu x

pilih=handles.pilih

switch handles.pilih %berhubungan dg pilihan pada
radiobutton

    case 1 %jika radiobutton1 yang ditekan

        y=A*sin(T*x+r); %maka menghitung nilai sinus

    case 2 %jika radiobutton2 yang ditekan

        y=A*cos(T*x+r); %maka menghitung nilai cosi
nus

end

axes(handles.axes1);

a=0; %membuat garis, seperti sumbu x dan sumbu y

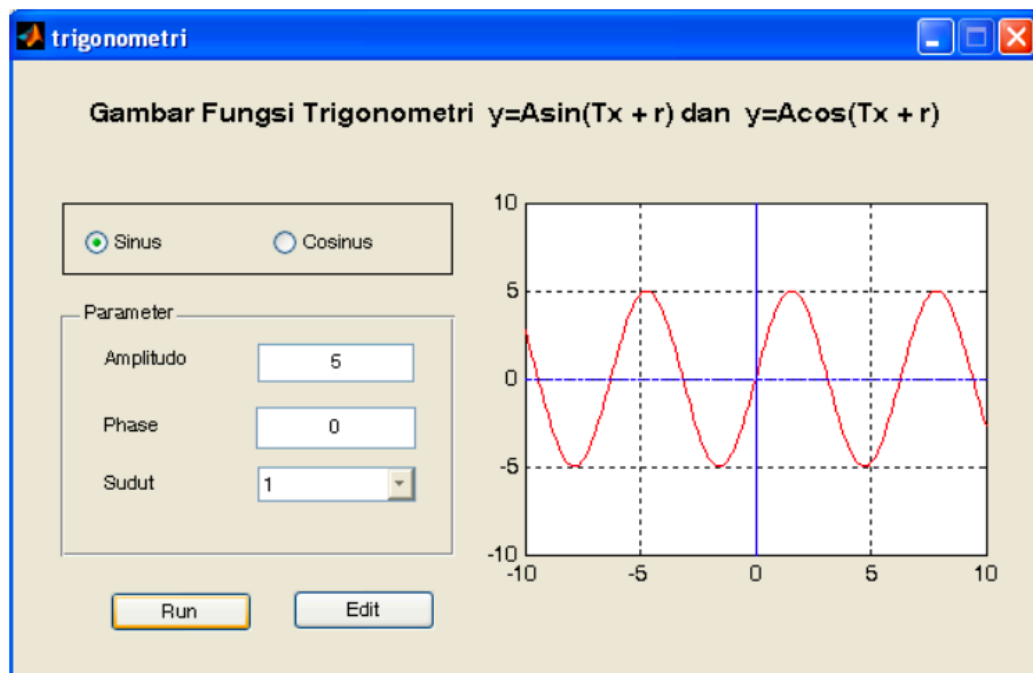
plot(x,y,'r', a, x, 'b', x, a, 'b');

grid on

```

#### d) Running GUI

Setelah langkah-langkah diatas dijalankan, langkah terakhir adalah menjalankan aplikasi yang telah dibuat dengan mengklik tombol Run dari jendela figure atau dari jendela debug m-file (tekan F5), sehingga akan muncul tampilan berikut.



*Gambar 10 Tampilan akhir GUI fungsi trigonometri*

## **Praktikum**

### Latihan 1

Buat aplikasi untuk dapat menampilkan grafik suatu persamaan polynomial pangkat tertentu yang di masukan kedalam inputan!

## **Laporan**

Buatlah laporan dalam buku kampus dengan menempelkan gambar hasil pengerjaan praktikum beserta penjelasan dengan tulisan tangan dari setiap pengerjaan yang dilakukan.

## **Referensi:**

1. <http://blog.pointopoin.com/>
2. Kasiman Peranginangin, "Pengenalan MATLAB", Penerbit ANDI, Yogyakarta 2006.
3. R.H. Sianipar, "Simulink Matlab Belajar dari Contoh", Penerbit ANDI 2015
4. <https://teuinsuska2009.files.wordpress.com/2010/10/modul-guideuploader-by-teuinsuska2009-wordpress-com.pdf>