NUMBER GUESSING GAME

&

ARITHMETIC CALCULATOR

&

BODY MASS INDEX CALCULATOR

NAME- DIPRIT KHAITAN

DEPARTMENT- CSE(AI)

SECTION - AI

CLASS ROLL NO - 29

ENROLLMENT NO – 12021002020031

SUBJECT - IT WORKSHOP LAB SUBJECT CODE - PCCCS392

INTRODUCTION

MATLAB is a powerful programming language that makes working with different mathematical equations easier. It is widely preferred by engineers and scientists. In this project, we will build a GUI-based simple calculator in MATLAB, which will take input and will return a value.

ARITHMETIC CALCULATOR - Calculators are one of the widely used instruments or applications in the world and using MATLAB we can design a simple calculator easily with basic functions .

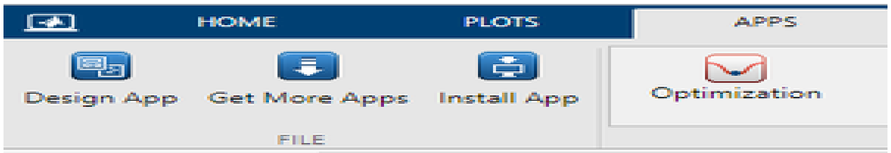
BMI CALCULATOR - Body mass index is a value derived from the mass and height of a person. The BMI is defined as the body mass divided by the square of the body height, and is expressed in units of kg/m², resulting from mass in kilograms and height in meters .

We will design the calculator which tells the BMI of the user according to the entered data .

ALGORITHM/PSEUDOCODE OF IMPLEMENTATION

- **Arithmetic Calculator :**

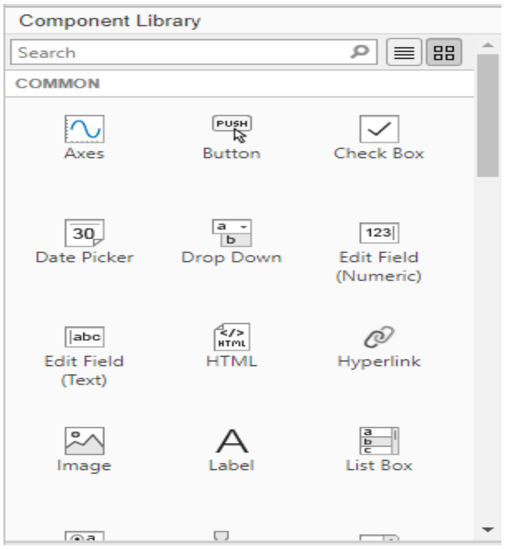
Step 1: To start working with the app, Open the editor, click on the Apps menu, and then on the Design App option.



The app designer, the workspace will be opened. Here you can choose templates for your app. We are going to implement it from scratch, so go Blank App option.

Now, our working space will be opened. Let's understand our workspace first before moving forward with our app.

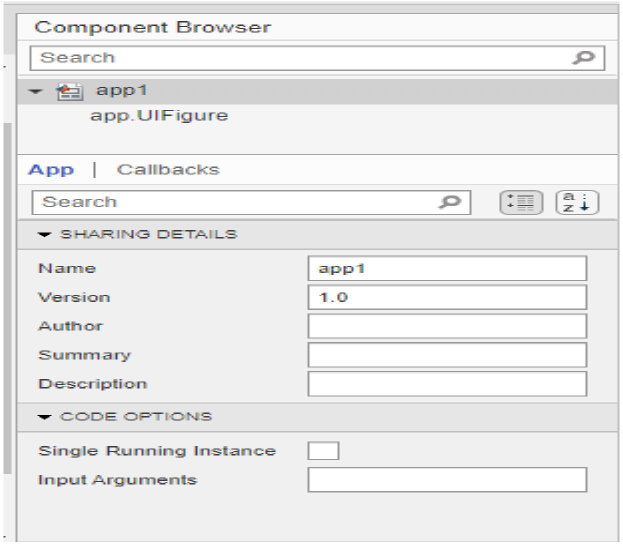
Step 2: In the leftmost part of the workspace, you can find the Component Library, you can select any of the components you want to insert in your app. You just need to drag the component and drop it in the workspace.



Step 3: Workspace IN the center you can find the workspace. It reflects the app you are working on. It consists of two parts:

- **Design View: You can design your app from here, like the positioning of different components, etc.**
- **Code View: You can make your app functional by adding a few lines of code to it. All the code that is added, is done from here.**

Step 4: Component Browser When you insert any component to your app and want to study or change its property, then you can do it from here. It controls naming to the description of the component in it.



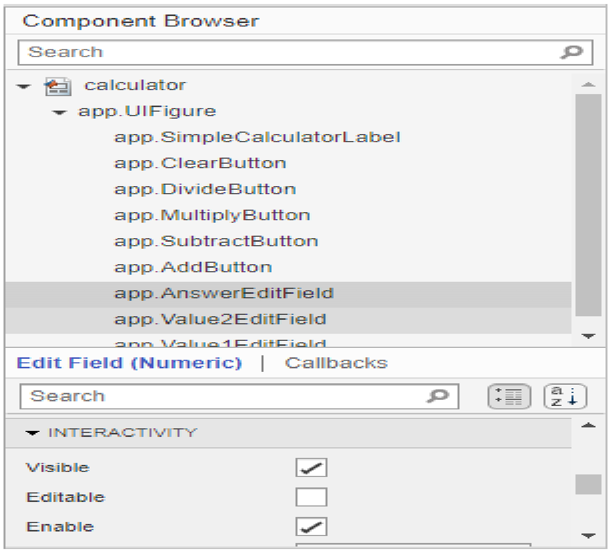
Name	app1
Version	1.0
Author	
Summary	
Description	
Single Running Instance	□
Input Arguments	

Step 5: Now to make the interface of your app follow these:

- **Drag and drop, 3 Edit Fields(Numeric), for storing two values and one for the answer.**
- **Now rename and arrange them using drag and drop.**
- **Now insert 7 buttons, namely for addition, subtraction, multiplication,**

division, square, square, and a clear button. We are going to work on these all.

- Add a label to the app for a better look.
- Also in the component Browser, search Interactivity for Answer_field and turn off editable.
- Arrange them with your preferred design, here we have designed it



Component Browser

Search

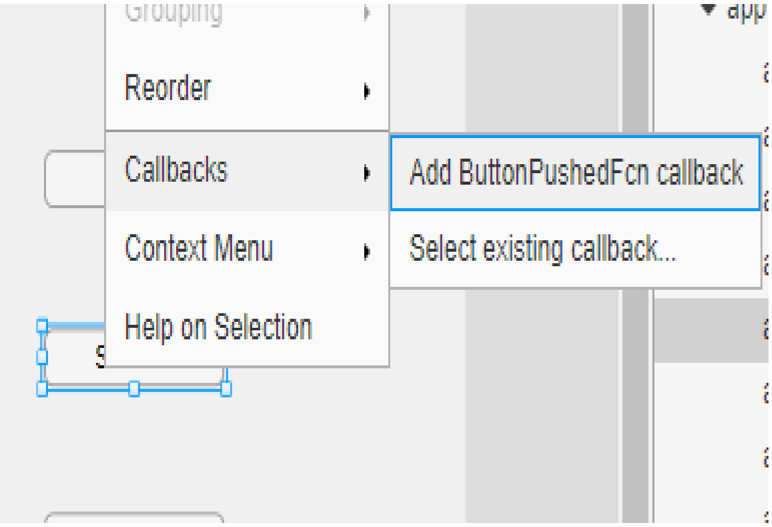
- calculator
  - app.UIFigure
    - app.SimpleCalculatorLabel
    - app.ClearButton
    - app.DivideButton
    - app.MultiplyButton
    - app.SubtractButton
    - app.AddButton
    - app.AnswerEditField
    - app.Value2EditField
    - app.Value1EditField

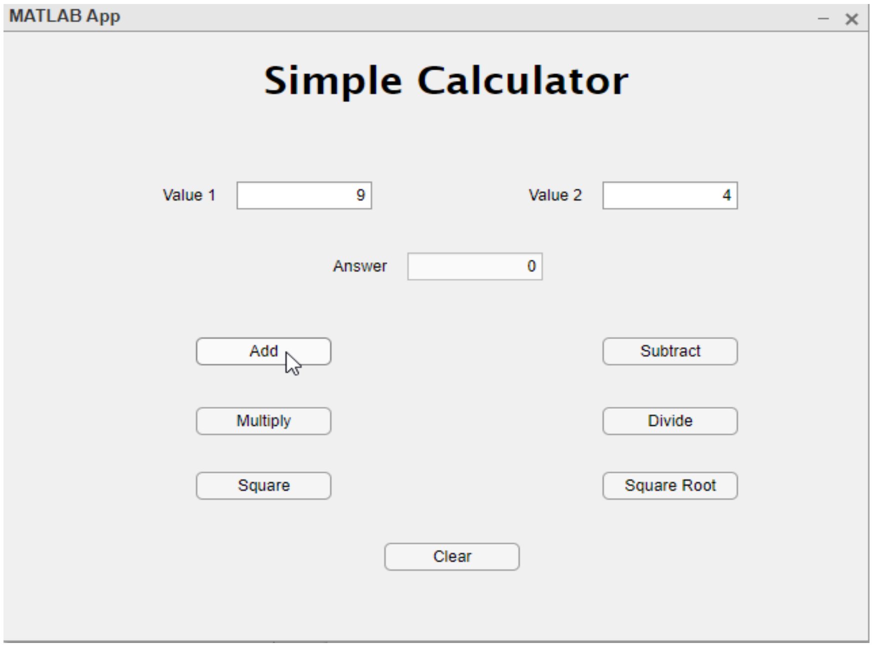
Edit Field (Numeric) | Callbacks

Search

INTERACTIVITY

Visible ☑
Editable □
Enable ☑





MATLAB App

Simple Calculator

Value 1: 9   Value 2: 4

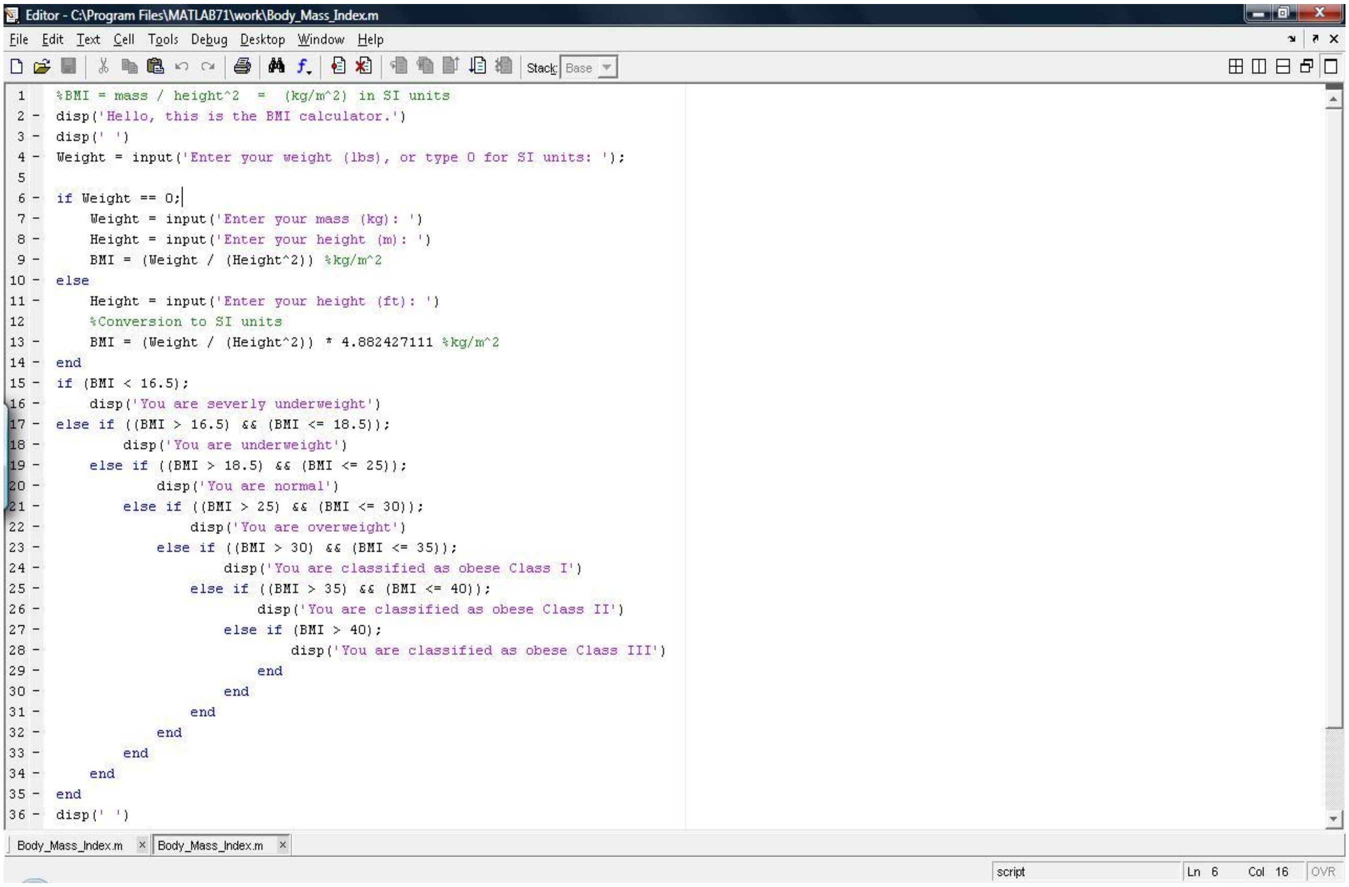
Answer: 0

Step 6: Now the interface of our app is ready. For adding code for any of the buttons, right-click on the button, select callback, add a new pushed_back_func.

- **BMI CALCULATOR**

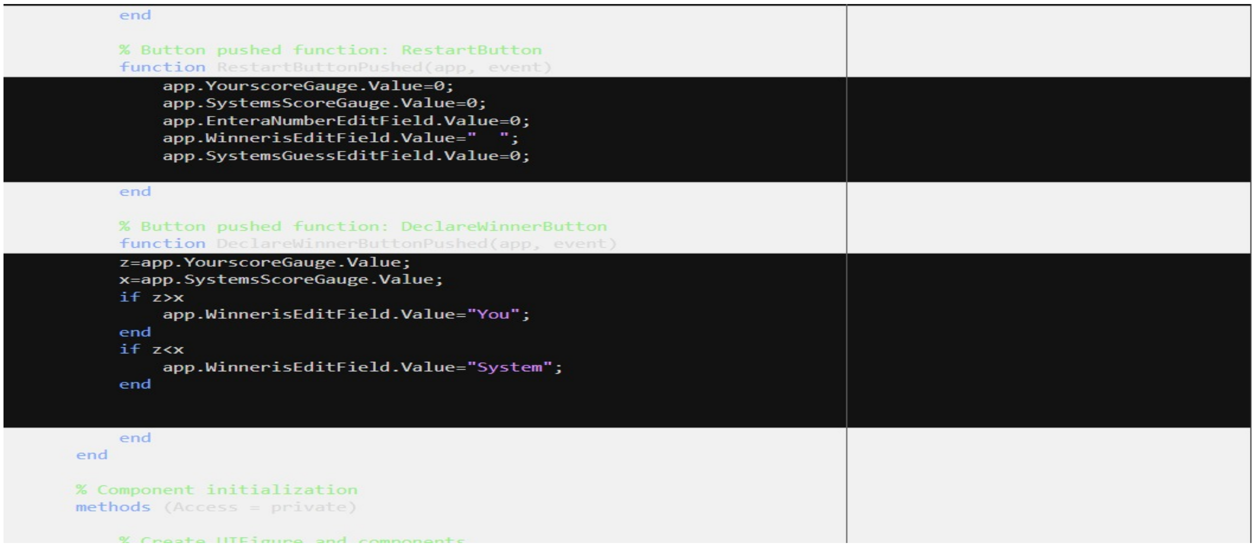
  CODE :

```
%BMI = mass / height^2  =  (kg/m^2) in SI units
disp('Hello, this is the BMI calculator.')
disp(' ')
Weight = input('Enter your weight (lbs), or type 0 for SI units: ');
if Weight == 0;
   Weight = input('Enter your mass (kg): ')
   Height = input('Enter your height (m): ')
   BMI = (Weight / (Height^2)) %kg/m^2
else
   Height = input('Enter your height (ft): ')
   %Conversion to SI units
   BMI = (Weight / (Height^2)) * 4.882427111 %kg/m^2
end
if (BMI < 16.5);
   disp('You are severly underweight')
else if ((BMI > 16.5) && (BMI <= 18.5));
       disp('You are underweight')
   else if ((BMI > 18.5) && (BMI <= 25));
           disp('You are normal')
       else if ((BMI > 25) && (BMI <= 30));
               disp('You are overweight')
           else if ((BMI > 30) && (BMI <= 35));
                   disp('You are classified as obese Class I')
               else if ((BMI > 35) && (BMI <= 40));
                       disp('You are classified as obese Class II')
                   else if (BMI > 40);
                           disp('You are classified as obese Class III')
                       end
                   end
               end
           end
       end
   end
end
disp(' ')
disp('Posssible BMI ranges: < 16.5 severe underweight, 16.5-18.5 underweight, 18.5-25 normal, 25-30 overweight, > 30 obese')
% The possible BMI categories are as follows: a BMI of less than 16.5
% indicates severe underweight, 16.5 to 18.5 is underweight, 18.5 to 25
% is normal, 25 to 30 is overweight. The Obese category range from 30-35
% (class I), 35-40 (class II), and over 40 is class III.
```

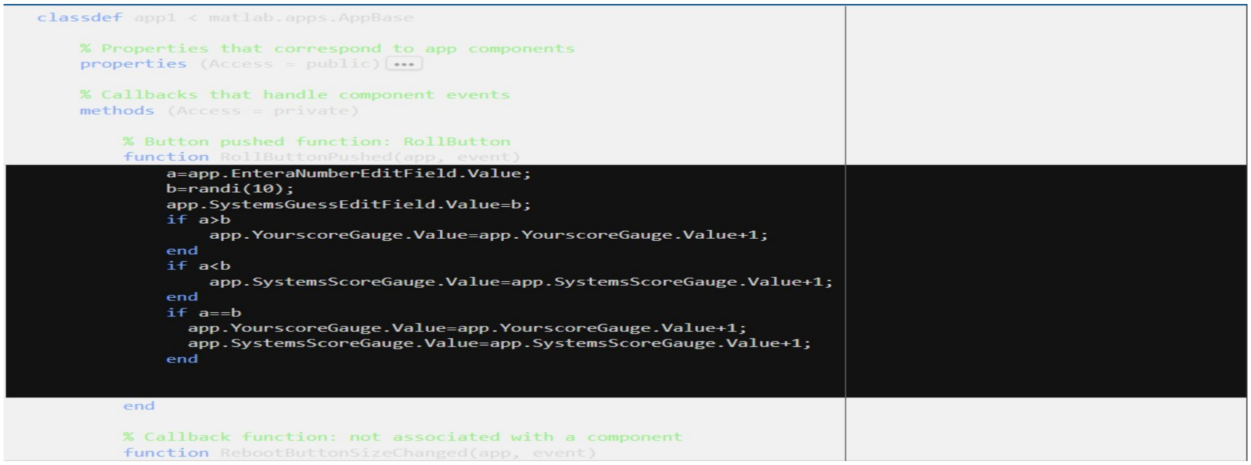

Editor - C:\Program Files\MATLAB71\work\Body_Mass_Index.m
File Edit Text Cell Tools Debug Desktop Window Help
Stack: Base
Body_Mass_Index.m
Body_Mass_Index.m
script
Ln 6
Col 16
OVR

```
%BMI = mass / height^2  =  (kg/m^2) in SI units
disp('Hello, this is the BMI calculator.')
disp(' ')
Weight = input('Enter your weight (lbs), or type 0 for SI units: ');

if Weight == 0;
    Weight = input('Enter your mass (kg): ')
    Height = input('Enter your height (m): ')
    BMI = (Weight / (Height^2)) %kg/m^2
else
    Height = input('Enter your height (ft): ')
    %Conversion to SI units
    BMI = (Weight / (Height^2)) * 4.882427111 %kg/m^2
end
if (BMI < 16.5);
    disp('You are severly underweight')
else if ((BMI > 16.5) && (BMI <= 18.5));
        disp('You are underweight')
    else if ((BMI > 18.5) && (BMI <= 25));
            disp('You are normal')
        else if ((BMI > 25) && (BMI <= 30));
                disp('You are overweight')
            else if ((BMI > 30) && (BMI <= 35));
                    disp('You are classified as obese Class I')
                else if ((BMI > 35) && (BMI <= 40));
                        disp('You are classified as obese Class II')
                    else if (BMI > 40);
                            disp('You are classified as obese Class III')
                        end
                    end
                end
            end
        end
    end
end
disp(' ')
```

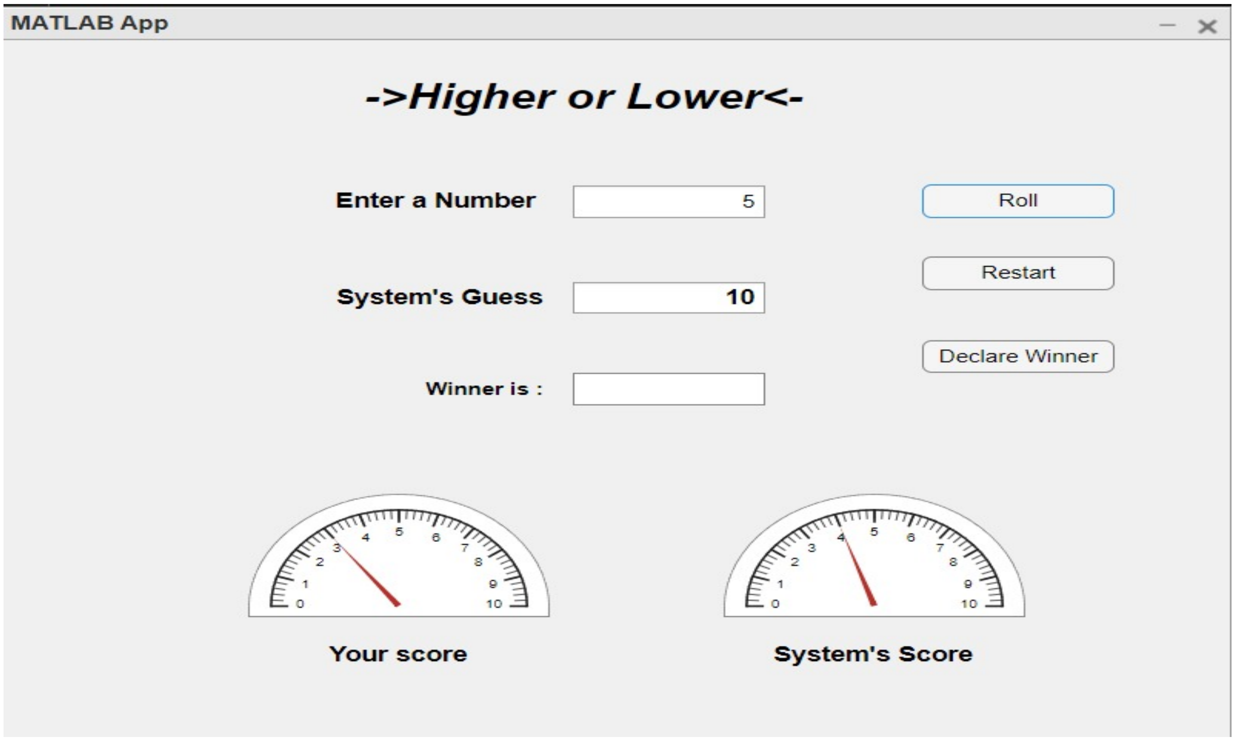
- **NUMBER GUESSING GAME :**



```
        end

        % Button pushed function: RestartButton
        function RestartButtonPushed(app, event)
            app.YourscoreGauge.Value=0;
            app.SystemsScoreGauge.Value=0;
            app.EnteraNumberEditField.Value=0;
            app.WinnerisEditField.Value="  ";
            app.SystemsGuessEditField.Value=0;

        end

        % Button pushed function: DeclareWinnerButton
        function DeclareWinnerButtonPushed(app, event)
        z=app.YourscoreGauge.Value;
        x=app.SystemsScoreGauge.Value;
        if z>x
            app.WinnerisEditField.Value="You";
        end
        if z<x
            app.WinnerisEditField.Value="System";
        end


        end
    end

    % Component initialization
    methods (Access = private)

        % Create UIFigure and components
```



```
classdef app1 < matlab.apps.AppBase

    % Properties that correspond to app components
    properties (Access = public) ...

    % Callbacks that handle component events
    methods (Access = private)

        % Button pushed function: RollButton
        function RollButtonPushed(app, event)
            a=app.EnteraNumberEditField.Value;
            b=randi(10);
            app.SystemsGuessEditField.Value=b;
            if a>b
                app.YourscoreGauge.Value=app.YourscoreGauge.Value+1;
            end
            if a<b
                app.SystemsScoreGauge.Value=app.SystemsScoreGauge.Value+1;
            end
            if a==b
              app.YourscoreGauge.Value=app.YourscoreGauge.Value+1;
              app.SystemsScoreGauge.Value=app.SystemsScoreGauge.Value+1;
            end

        end

        % Callback function: not associated with a component
        function RebootButtonSizeChanged(app, event)
```



MATLAB App

->Higher or Lower<-

Enter a Number: 5

System's Guess: 10

Winner is :

CONCLUSION

So I will finally conclude this project . It provided me with great insights and helped to learn how to write functions as well as use the apps feature in MATLAB which is pretty helpful and made app making certainly much faster and efficient .