



**LAB-5**

**Course:**

**Internet Application Development(CIS-305L)**

**Submitted to:**

**Mr. Irfan Hameed**

**Submitted by:**

**Asad Butt**

**Haider Ali Aurangzaib**

**Muhammad Ahtasham**

**Muhammad Rizwan Rizvi**

**Zain Abbas**

**Dated:**

**March 24, 2024**

**Department of Computer and Information Science, PIEAS**

## Laboratory Module 5

### Part A)

Using Html server controls develop a web form that displays a rectangle {refer to book page 148}. Write appropriate event handler code that detects the location at which the user has clicked.

### Part B)

Modify code developed in Part (A) to identify regions on the following circle:

[Hint:] Use Screen coordinates, Cartesian coordinates and Polar coordinates

#### **.aspx code (part-1):**

```
<%@ Page Title="Home Page" Language="VB" Codefile="Default.aspx.vb"
Inherits="_Default" %>
<!DOCTYPE html>
<html lang="en">
<head>
  <meta charset="UTF-8">
  <meta name="viewport" content="width=device-width, initial-scale=1.0">
  <title>Button Click Detector</title>
  <style>
    #rectangle {
      border: 5px solid red; /* Add border to the button */
      padding: 15px; /* Add padding to separate border from content */
      border-radius: 5%; /* Make the button circular */
    }
  </style>
</head>
<body>
  <form id="form1" runat="server">
    <div>
      <asp:ImageButton ID="rectangle" runat="server"
        Width="300px" Height="150px" BackColor="LightBlue"
        Style="cursor: pointer;" OnClick="rectangle_Click" />
      <asp:Label ID="lblClickLocation" runat="server" Text=""></asp:Label>
      <asp:Label ID="lblClickArea" runat="server" Text=""></asp:Label>
    </div>
  </form>
</body>
</html>
```

#### **.aspx.vb code(part-1):**

```
Imports System.Drawing ' Required for PointF

Public Class _Default
  Inherits Page
```

```

Protected Sub Page_Load(ByVal sender As Object, ByVal e As EventArgs) Handles Me.Load

End Sub

Protected Sub rectangle_Click(ByVal sender As Object, ByVal e As ImageClickEventArgs)
Handles rectangle.Click
    Dim imageX = Rectangle.Width.Value / 2 ' Get button center X
    Dim imageY = Rectangle.Height.Value / 2 ' Get button center Y
    Dim clickX = e.X
    Dim clickY = e.Y

    ' Calculate distance from click to button center
    Dim distance = Math.Sqrt(CDec(Math.Pow(clickX - imageX, 2)) +
CDec(Math.Pow(clickY - imageY, 2)))

    If distance <= Rectangle.Width.Value / 2 Then ' Click within button radius
(including border)
        lblClickArea.Text = "<br>Clicked inside the button"
        lblClickLocation.Text = "<br>Clicked at (" & clickX.ToString() & ", " &
clickY.ToString() & ")"
    Else
        lblClickArea.Text = "<br>Clicked outside the button"
        lblClickLocation.Text = ""
    End If
End Sub
End Class

```

**.aspx code (part-2):**

```

<%@ Page Language="VB" AutoEventWireup="false" CodeFile="Default2.aspx.vb"
Inherits="Default2" %>

<!DOCTYPE html>

<html xmlns="http://www.w3.org/1999/xhtml">
<head runat="server">
    <title>Input image</title>
    <style>
        #ImgButton{
            width: 200px;
            height: 200px;
            border-radius: 50%;
        }
    </style>
</head>
<body>
    <form id="form1" runat="server">
        <div>
            <input type="image" ID="ImgButton" runat="server" src="circle.png" />
        </div>
        <asp:Label ID="Result" runat="server" Text=""></asp:Label>
    </form>
</body>
</html>

```

**.aspx.vb code (part-2):**

```

Partial Class Default2

```

```

Inherits System.Web.UI.Page
Protected Sub ImgButton_ServerClick(ByVal sender As Object, ByVal e As
ImageClickEventArgs) Handles ImgButton.ServerClick

    Dim x As Integer = Request.Form("ImgButton.x")
    Dim y As Integer = Request.Form("ImgButton.y")

    ' Calculate distance from the center of the circle
    Dim centerX As Integer = 100 ' x-coordinate of the center
    Dim centerY As Integer = 100 ' y-coordinate of the center
    Dim distance As Double = Math.Sqrt((x - centerX) ^ 2 + (y - centerY) ^ 2)

    ' Calculate angle in degrees
    Dim angle As Double = Math.Atan2(y - centerY, x - centerX) * (180 / Math.PI)
    If angle < 0 Then
        angle += 360 ' Ensure positive angle
    End If

    ' Determine the region based on distance and angle
    Dim region As String = ""
    If distance <= 100 AndAlso angle >= 180 AndAlso angle < 270 Then
        region = "Red"
    ElseIf distance <= 100 AndAlso angle >= 270 AndAlso angle < 360 Then
        region = "Blue"
    ElseIf distance <= 100 AndAlso angle >= 90 AndAlso angle < 180 Then
        region = "Green"
    Else
        region = "Yellow"
    End If

    ' Display the output
    Result.Text = "Clicked at (" & x & ", " & y & ") in " & region & " region at
angle: " & angle

End Sub
End Class

```

OUTPUTS:

