CSNB544/CSNB5123 Mobile Application Development (2023)

Lab 3 – Xamarin.Form Layout

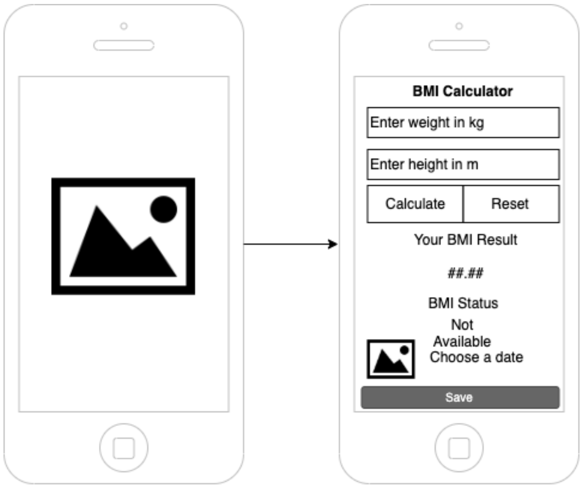
Student ID: SW0107187	Student Name: Akmal Arif Bin Arifin

Instruction: Answer all questions. Write your answer in the table located at the final page of this document.

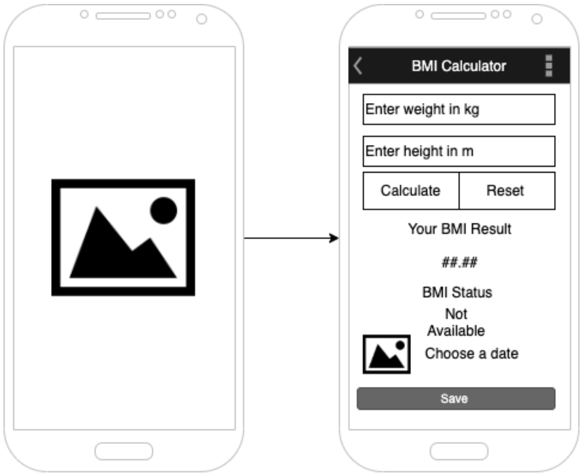
You are going to learn to develop a cross platform application, BMI Calculator. This application will receive two inputs from the user. Based on these inputs, it will calculate the BMI value. Then the application will identify which group that the BMI result belongs to. The application should be able to store the calculated result.

Lab 3 until Lab 7 will guide you how to develop this app. Please follow the lab recordings stored on teams for better guidance and understanding.

iOS



Android

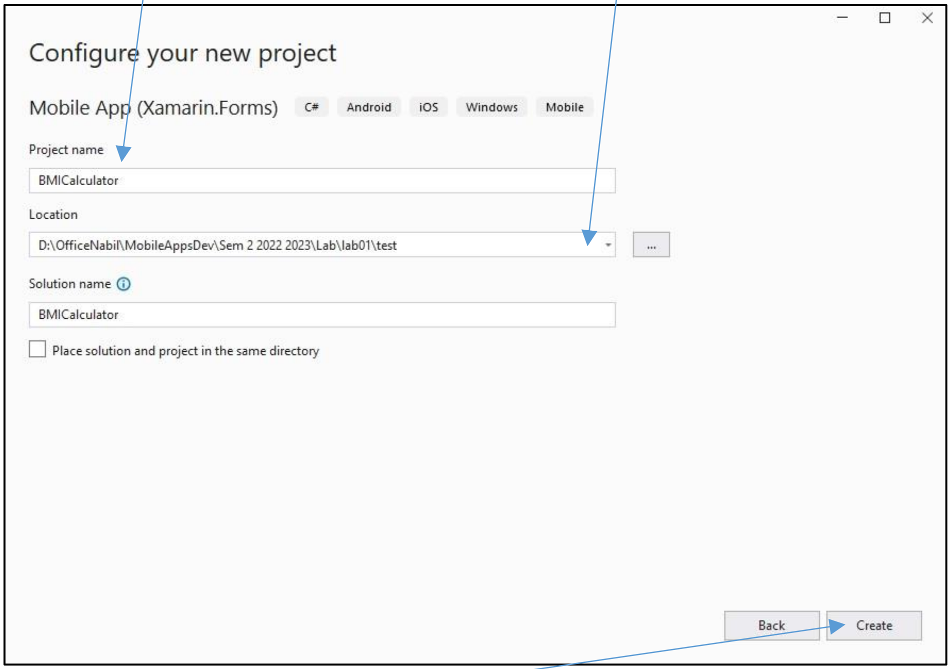


Student ID: SW0107187	Student Name: Akmal Arif Bin Arifin

Activity 1 – Create a Cross Platform application

You need to screen capture each of the following highlighted activities.

1. First start Visual Studio
2. Create a new project.
3. For Windows, choose: Mobile App (Xamarin.Forms)
4. For MacOS: Multiplatform -> App -> Blank Forms App
5. Next
6. Project Name: **BMICalculator**
7. For location, choose a folder that you will use to save this project.



Configure your new project

Mobile App (Xamarin.Forms)

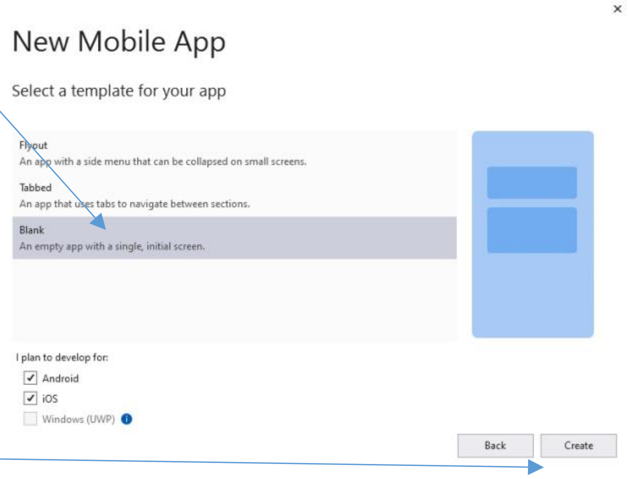
Project name
BMICalculator

Location
D:\OfficeNabil\MobileAppsDev\Sem 2 2022 2023\Lab\lab01\test

Solution name
BMICalculator

□ Place solution and project in the same directory

8. Create
9. Select a template: Blank
10. Check: Android and iOS
11. Create



New Mobile App

Select a template for your app

Flyout
An app with a side menu that can be collapsed on small screens.

Tabbed
An app that uses tabs to navigate between sections.

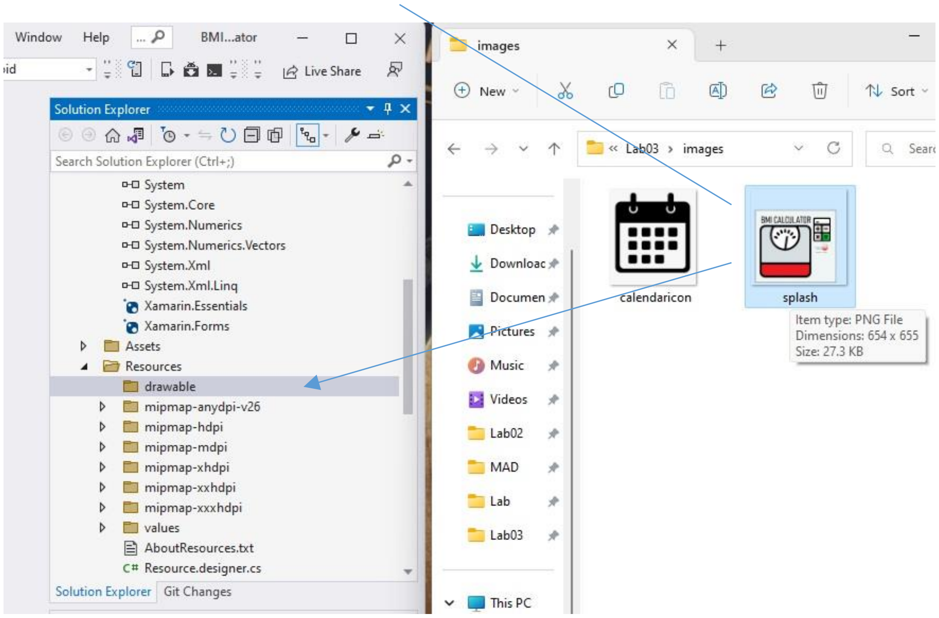
Blank
An empty app with a single, initial screen.

I plan to develop for:
☑ Android
☑ iOS
□ Windows (UWP)

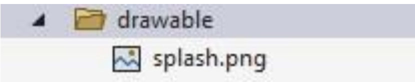
12. For MacOS: Check Shared Code: Use .NET Standard
13. For MacOS: Do not rename the project name and solution name.

Activity 2 – Splash screen on Android

1. Download splash.png from our lab 3 resources on Brighten.
2. In the solution explorer, look for Resources > drawable.
3. Click and drag the splash.png into drawable folder.



   a. After dragging it, you will see that the splash.png will be inside your drawable folder.



4. Next, on the solution explorer **Expand Resources > values** and then double click on **styles.xml.**
5. Add the following highlighted code into this file before the </resources> line

```
<resources>
   <style name="MainTheme" parent="MainTheme.Base">
   </style>
   ...
   <style name="AppCompatDialogStyle" parent="Theme.AppCompat.Light.Dialog">
      <item name="colorAccent">#FF4081</item>
   </style>
   <style name="Splash" parent="android:Theme">
      <item name="android:windowBackground">@drawable/splash</item>
      <item name="android:windowNoTitle">true</item>
   </style>
</resources>
```

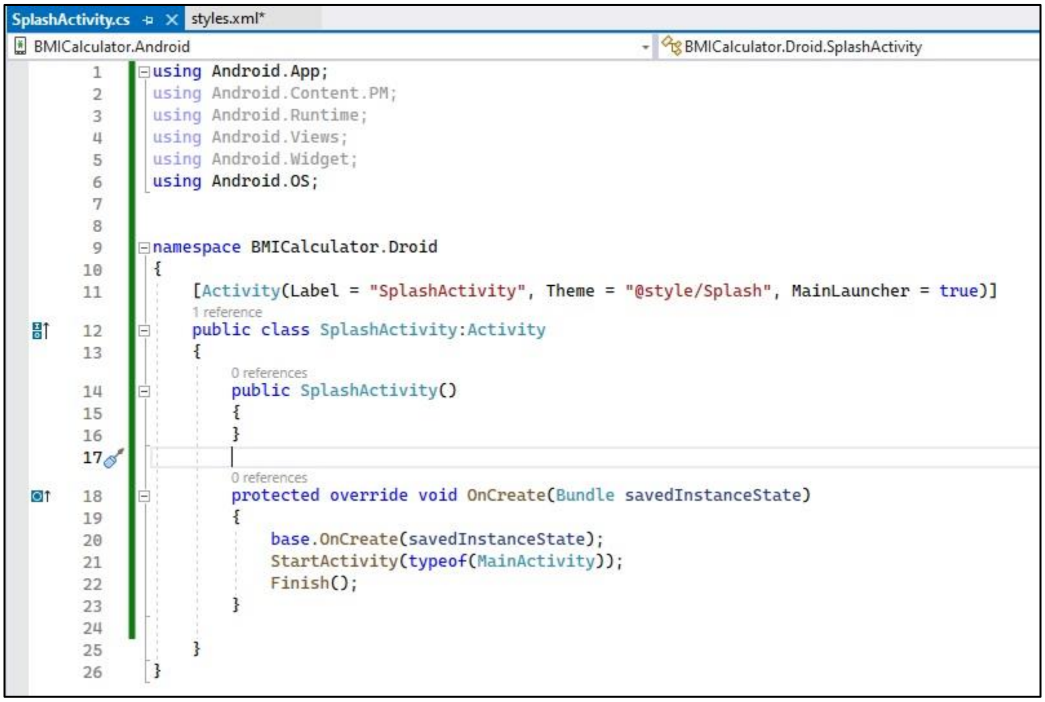
6. Create an activity, SplashActivity.cs by right clicking on BMICalculator.Android > Add > Class... > for the name: SplashActivity.cs > Click Add
    a. Add activity details as highlighter below and set this activity as the MainLauncher. You need to implement this activity as Activity.

```
using System;
using Android.App;
using Android.Content.PM;
using Android.Runtime;
using Android.Views;
using Android.Widget;
using Android.OS;

….
Namespace BMICalculator.Droid
{
[Activity(Label = "SplashActivity", Theme = "@style/Splash", MainLauncher = true)]
    public class SplashActivity:Activity
    {
        public SplashActivity()
        {
        }
    }
```

7. Next, add the OnCreate function and then start the MainActivity.

```
[Activity(Label = "SplashActivity", Theme = "@style/Splash", MainLauncher = true)]
    public class SplashActivity:Activity
    {
        public SplashActivity()
        {
        }
        protected override void OnCreate(Bundle savedInstanceState)
        {
            base.OnCreate(savedInstanceState);
            StartActivity(typeof(MainActivity));
            Finish();
        }
    }
```

8. Your finished SplashActivity.cs should look like the one below:



```
using Android.App;
using Android.Content.PM;
using Android.Runtime;
using Android.Views;
using Android.Widget;
using Android.OS;


namespace BMICalculator.Droid
{
    [Activity(Label = "SplashActivity", Theme = "@style/Splash", MainLauncher = true)]
    public class SplashActivity:Activity
    {
        public SplashActivity()
        {
        }

        protected override void OnCreate(Bundle savedInstanceState)
        {
            base.OnCreate(savedInstanceState);
            StartActivity(typeof(MainActivity));
            Finish();
        }

    }
}
```

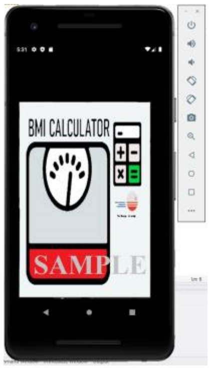
9. Next, open MainActivity.cs from the solution explorer
10. Remove the MainLauncher = true from the MainActivity.cs file.

```
    [Activity(Label = "BMICalculator1", Icon = "@mipmap/icon", MainLauncher = true ,Theme =
"@style/MainTheme", ConfigurationChanges = ConfigChanges.ScreenSize |
ConfigChanges.Orientation)]
```

11. Next, find the AndroidManifest.xml in the solution exporer. We need to declare the SplashActivity and MainActivity
12. Go to BMICalculator.Android > Properties > AndroidManifest.xml. Double click on AndroidManifest.xml and add the highlighted codes.

```
<application android:label="BMICalculator.Android">
        <activity android:name=".SplashActivity"></activity>
        <activity android:name=".MainActivity"></activity>
    </application>
```

13. Save and Run this application on Android emulator. It will take some time to load the first time. Wait until you see the splash screen.
14. Screen capture the Splash Screen on Android.
15. **Remember to stop execution before you start editing again.**

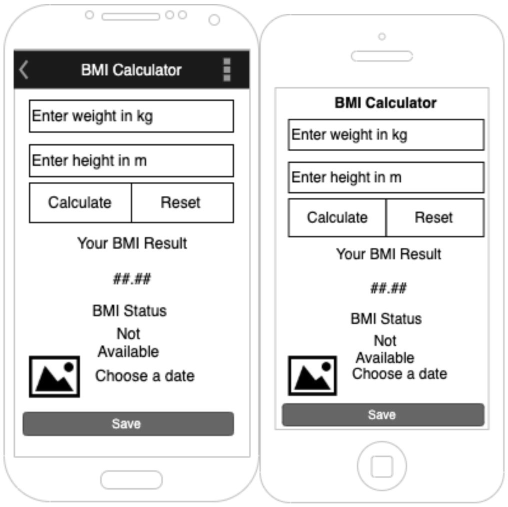


Activity 3 – Splash screen on iOS (You may skip this if you are on windows)

1. Expand BMICalculator.iOS
2. Click Assets.xcassets
3. Click plus symbol icon (add new assets)
4. Choose New Image Set
5. Double click the set to rename it as splash
6. Drag & drop the image files into the first row of placeholder boxes according to the resolutions defined by Apple.
7. Click and drag the splash image into Universal 1x, 2x and 3x
8. Finally, what's left is to update the Resources > LaunchScreen.storyboard to include our new image. On the toolbox, click and drag ImageView into the centre of the screen.
9. Click Properties. ImageView > Image > Choose Splash.
10. To change the background color to white, click LaunchScreen.storyboard
11. Click anywhere on the blue screen Click Properties tab > View > Background > White color and Tint > White Color
12. MacOS only: Run this application on iOS emulator.

Activity 4 – Add Entry to apps

1. Refer to the following activities on Android and iOS. In this section, we will draw the top two views, Entry elements and layouts behind it.



2. In the MainPage.xaml, remove all the content located witihin <StackLayout> element.

```
<?xml version="1.0" encoding="utf-8"?>
<ContentPage ...    >
   <StackLayout>

   </StackLayout>
</ContentPage>
```

3. Add another <StackLayout> within the existing <StackLayout>

```
<?xml version="1.0" encoding="utf-8"?>
<ContentPage ...    >
   <StackLayout>
      <StackLayout>

      </StackLayout>
   </StackLayout>
</ContentPage>
```

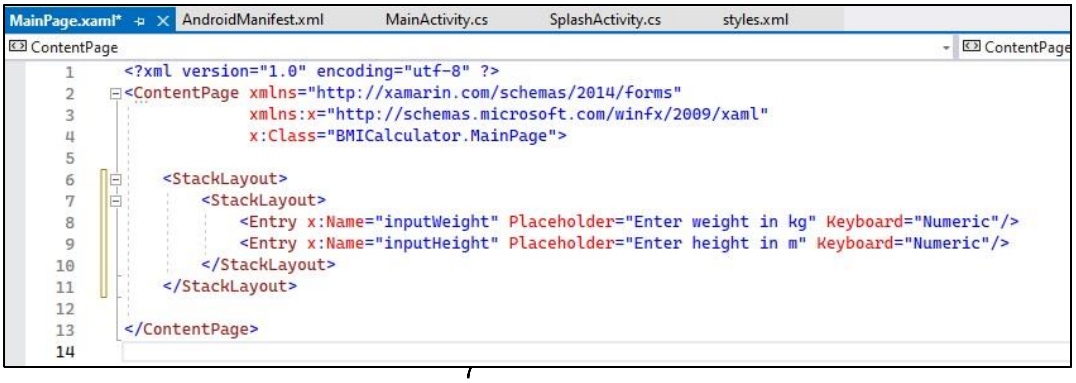
4. Add an input field from the user using <Entry>. Add the following attributes of this element.
   a. Assign the name as inputWeight.
   b. Add a hint of this entry field using Placeholder with value “Enter weight in kg”
   c. Set the keyboard for input value to numeric.

```
<?xml version="1.0" encoding="utf-8"?>
<ContentPage ...    >
   <StackLayout>
      <StackLayout>
            <Entry x:Name="inputWeight" Placeholder="Enter weight in kg"
Keyboard="Numeric"/>
      </StackLayout>
   </StackLayout>
</ContentPage>
```

5. Add an input field from the user using <Entry>. Add the following attributes of this element.
   a. Assign the name as inputHeight.
   b. Add a hint of this entry field using Placeholder with value “Enter height in m”
   c. Set the keyboard for input value to numeric.

```
<?xml version="1.0" encoding="utf-8"?>
<ContentPage ...    >
   <StackLayout>
      <StackLayout>
            <Entry x:Name="inputWeight" Placeholder="Enter weight in kg"
Keyboard="Numeric"/>
         <Entry x:Name="inputHeight" Placeholder="Enter height in m" Keyboard="Numeric"/>
      </StackLayout>
   </StackLayout>
</ContentPage>
```

6. Your current MainPage.xaml should look like the one below:



```
<?xml version="1.0" encoding="utf-8" ?>
<ContentPage xmlns="http://xamarin.com/schemas/2014/forms"
             xmlns:x="http://schemas.microsoft.com/winfx/2009/xaml"
             x:Class="BMICalculator.MainPage">

    <StackLayout>
        <StackLayout>
            <Entry x:Name="inputWeight" Placeholder="Enter weight in kg" Keyboard="Numeric"/>
            <Entry x:Name="inputHeight" Placeholder="Enter height in m" Keyboard="Numeric"/>
        </StackLayout>
    </StackLayout>

</ContentPage>
```

7. Run this application on Android/iOS emulator. Screen capture this activity.

Activity 5 – Set safe area to iOS platform (You may skip this if you are on windows)

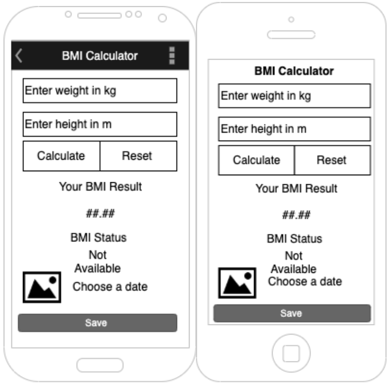
1. On iOS, you may find the Entry elements are positioned behind the front camera.
2. In the MainPage.xaml, implement the iOS safe area. Add the following highlighted code.

```
<ContentPage
    xmlns="http://xamarin.com/schemas/2014/forms"
    xmlns:x="http://schemas.microsoft.com/winfx/2009/xaml"
    xmlns:d="http://xamarin.com/schemas/2014/forms/design"
    xmlns:mc="http://schemas.openxmlformats.org/markup-compatibility/2006"
    mc:Ignorable="d"
    x:Class="BMICalculatorForm.MainPage"
    xmlns:ios="clr-
namespace:Xamarin.Forms.PlatformConfiguration.iOSSpecific;assembly=Xamarin.Forms.
Core"
    ios:Page.UseSafeArea="true"
>
```

3. Run this application on iOS emulator. Screen capture the BMI Calculator activity.

Activity 6 – Add Button to the apps

1. Refer to the following activities on Android and iOS. In this section, we will draw the next two views, Button elements and layouts behind it.



2. We will use FlexLayout to arrange these two buttons side by side within a single row.

```
<StackLayout >
            <Entry x:Name="inputWeight" Placeholder="Enter weight in kg"
Keyboard="Numeric"/>
            <Entry x:Name="inputHeight" Placeholder="Enter height in m" Keyboard="Numeric"/>
</StackLayout>
        <FlexLayout Direction="Row"
                AlignItems="Center"
                JustifyContent="SpaceEvenly">
        </FlexLayout>
```

3. Add a button, Calculate. Set this button name as calculateButton.

```
<FlexLayout Direction="Row"
            AlignItems="Center"
            JustifyContent="SpaceEvenly">
        <Button x:Name="calculateButton" Text="Calculate" VerticalOptions="CenterAndExpand" />
</FlexLayout>
```

4. Add a button, Reset. Set this button name as resetButton.

```
<FlexLayout Direction="Row"
            AlignItems="Center"
            JustifyContent="SpaceEvenly">
        <Button x:Name="calculateButton" Text="Calculate" VerticalOptions="CenterAndExpand" />
        <Button x:Name="resetButton" Text="Reset" VerticalOptions="CenterAndExpand"/>
</FlexLayout>
```

5. Set the width of each button to 110.

```
<FlexLayout Direction="Row"
            AlignItems="Center"
            JustifyContent="SpaceEvenly">
        <Button x:Name="calculateButton" Text="Calculate" VerticalOptions="CenterAndExpand"
WidthRequest="110" />
        <Button x:Name="resetButton" Text="Reset" VerticalOptions="CenterAndExpand"
WidthRequest="110" />
     </FlexLayout>
```

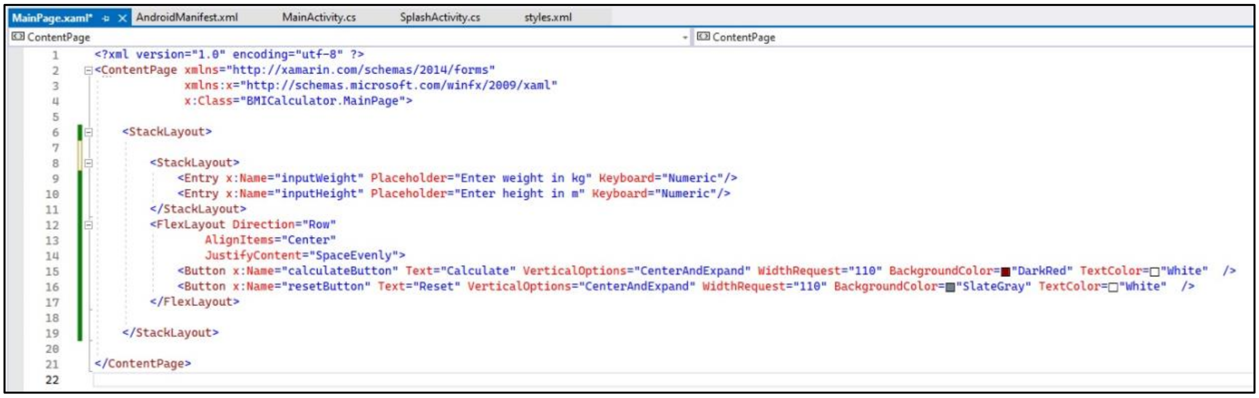
6. Change Button color using BackgroundColor. Use code “DarkRed” for Calculate button and “SlateGray” for Reset button.

```
<Button x:Name="calculateButton" Text="Calculate" VerticalOptions="CenterAndExpand"
WidthRequest="110" BackgroundColor="DarkRed" />
<Button x:Name="resetButton" Text="Reset" VerticalOptions="CenterAndExpand"
WidthRequest="110" BackgroundColor="SlateGray" />
```

7. Change the Button’s text color using TextColor. Use code “White” for Calculate button and Reset button.

```
<Button x:Name="calculateButton" Text="Calculate" VerticalOptions="CenterAndExpand"
WidthRequest="110" BackgroundColor="DarkRed" TextColor="White" />
<Button x:Name="resetButton" Text="Reset" VerticalOptions="CenterAndExpand"
WidthRequest="110" BackgroundColor="SlateGray" TextColor="White" />
```

8. The MainPage.xaml should now look like the one below.

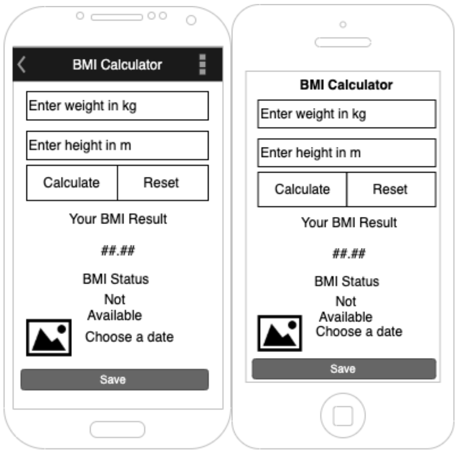


```
<?xml version="1.0" encoding="utf-8" ?>
<ContentPage xmlns="http://xamarin.com/schemas/2014/forms"
             xmlns:x="http://schemas.microsoft.com/winfx/2009/xaml"
             x:Class="BMICalculator.MainPage">

    <StackLayout>

        <StackLayout>
            <Entry x:Name="inputWeight" Placeholder="Enter weight in kg" Keyboard="Numeric"/>
            <Entry x:Name="inputHeight" Placeholder="Enter height in m" Keyboard="Numeric"/>
        </StackLayout>
        <FlexLayout Direction="Row"
                    AlignItems="Center"
                    JustifyContent="SpaceEvenly">
            <Button x:Name="calculateButton" Text="Calculate" VerticalOptions="CenterAndExpand" WidthRequest="110" BackgroundColor=■"DarkRed" TextColor=□"White"  />
            <Button x:Name="resetButton" Text="Reset" VerticalOptions="CenterAndExpand" WidthRequest="110" BackgroundColor=■"SlateGray" TextColor=□"White"  />
        </FlexLayout>

    </StackLayout>

</ContentPage>
```

9. Run the application. Screen capture the BMI Calculator activity.

Activity 7 – Add Label to the apps

1. Refer to the following activities on Android and iOS. In this section, we will draw the Label elements and layouts behind it.



2. We will use StackLayout to arrange these two buttons side by side within a single row.

```
<FlexLayout Direction="Row"
            AlignItems="Center"
            JustifyContent="SpaceEvenly"
            >
        ...
</FlexLayout>
<StackLayout HorizontalOptions="Center">

</StackLayout>
```

3. Add 4 Labels with Text value; Your BMI Result, 0.00, BMI Status, and Not Available.

```
<StackLayout HorizontalOptions="Center">
        <Label Text="Your BMI Result" HorizontalTextAlignment="Center"/>
        <Label Text="0.00" HorizontalTextAlignment="Center" />
        <Label Text="BMI Status" HorizontalTextAlignment="Center"/>
        <Label Text="Not Available" HorizontalTextAlignment="Center" />
</StackLayout>
```

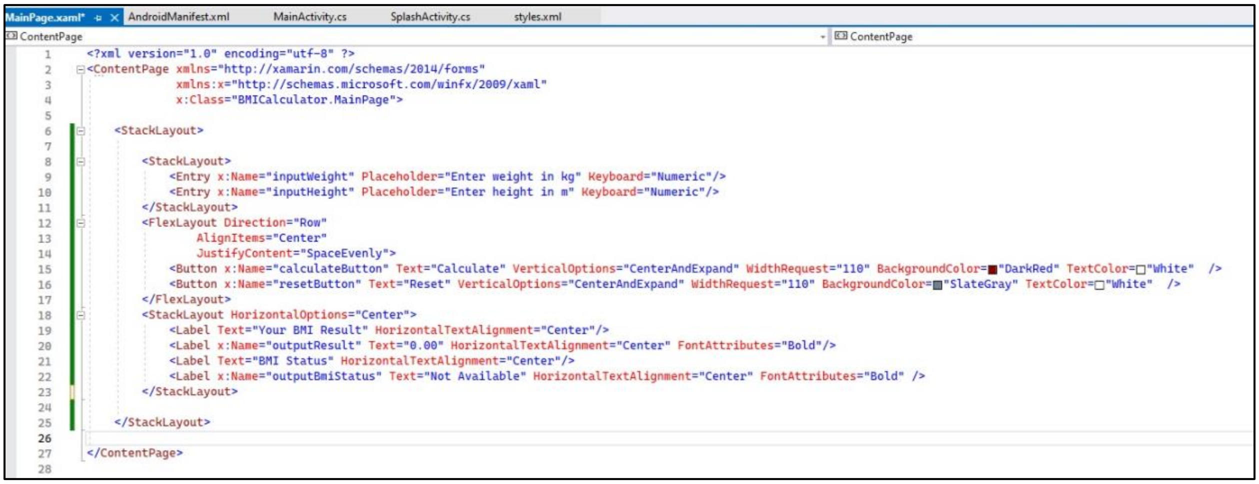
4. Assign a name to the text labeled “0.00”, outputResult by using x:Name, and text labeled “Not Available” with name outputBmiStatus.

```
<StackLayout HorizontalOptions="Center">
        <Label Text="Your BMI Result" HorizontalTextAlignment="Center"/>
        <Label x:Name="outputResult" Text="0.00" HorizontalTextAlignment="Center" />
        <Label Text="BMI Status" HorizontalTextAlignment="Center"/>
        <Label x:Name="outputBmiStatus" Text="Not Available" HorizontalTextAlignment="Center" />
</StackLayout>
```

5. Bold the outputResult Label and Not Available.

```
<StackLayout HorizontalOptions="Center">
            <Label Text="Your BMI Result" HorizontalTextAlignment="Center"/>
            <Label x:Name="outputResult" Text="0.00" HorizontalTextAlignment="Center"
FontAttributes="Bold"/>
            <Label x:Name="outputBmiStatus Text="BMI Status" HorizontalTextAlignment="Center"/>
            <Label Text="Not Available" HorizontalTextAlignment="Center" FontAttributes="Bold"/>
</StackLayout>
```

6. Your MainPage.xaml should have the following code:

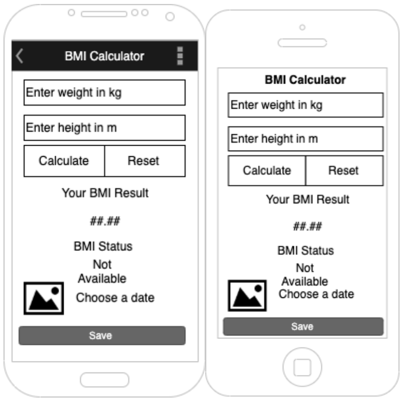


```
<?xml version="1.0" encoding="utf-8" ?>
<ContentPage xmlns="http://xamarin.com/schemas/2014/forms"
             xmlns:x="http://schemas.microsoft.com/winfx/2009/xaml"
             x:Class="BMICalculator.MainPage">

    <StackLayout>

        <StackLayout>
            <Entry x:Name="inputWeight" Placeholder="Enter weight in kg" Keyboard="Numeric"/>
            <Entry x:Name="inputHeight" Placeholder="Enter height in m" Keyboard="Numeric"/>
        </StackLayout>
        <FlexLayout Direction="Row"
                AlignItems="Center"
                JustifyContent="SpaceEvenly">
            <Button x:Name="calculateButton" Text="Calculate" VerticalOptions="CenterAndExpand" WidthRequest="110" BackgroundColor=■"DarkRed" TextColor=□"White"  />
            <Button x:Name="resetButton" Text="Reset" VerticalOptions="CenterAndExpand" WidthRequest="110" BackgroundColor=■"SlateGray" TextColor=□"White"  />
        </FlexLayout>
        <StackLayout HorizontalOptions="Center">
            <Label Text="Your BMI Result" HorizontalTextAlignment="Center"/>
            <Label x:Name="outputResult" Text="0.00" HorizontalTextAlignment="Center" FontAttributes="Bold"/>
            <Label Text="BMI Status" HorizontalTextAlignment="Center"/>
            <Label x:Name="outputBmiStatus" Text="Not Available" HorizontalTextAlignment="Center" FontAttributes="Bold" />
        </StackLayout>

    </StackLayout>

</ContentPage>
```

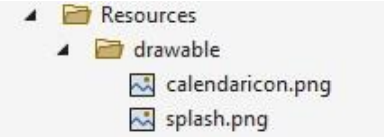
7. Run this application. Screen capture this activity.

Activity 8 – Add Image to the apps

1. Refer to the following activities on Android and iOS. In this section, we will add an Image elements and layouts behind it.



2. Download **calendaricon.png** from our lab 3 resources on Brighten.
3. For Android project, click and drag the downloaded picture into drawable folder just like activity 2 before - Resources > drawable.



Student ID: SW0107187	Student Name: Akmal Arif Bin Arifin

4. Add a new FlexLayout element after the previous StackLayout.

```
<FlexLayout Direction="Row"
            AlignItems="Center"
            JustifyContent="SpaceEvenly">

</FlexLayout>
```

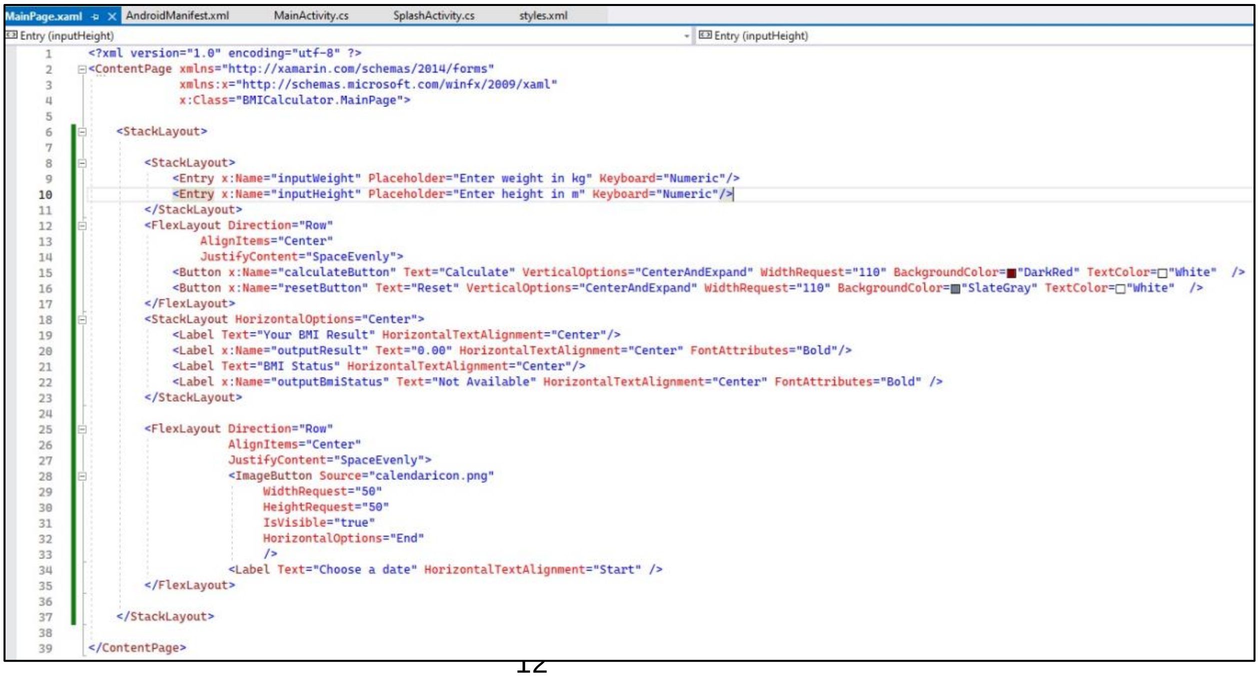
5. Add an ImageButton represents by the downloaded file, calendaricon.png. Preset the width and height to 50. Set the visible to true.

```
<FlexLayout Direction="Row"
            AlignItems="Center"
            JustifyContent="SpaceEvenly">
        <ImageButton Source="calendaricon.png"
                WidthRequest="50"
                HeightRequest="50"
                IsVisible="true"
                HorizontalOptions="End"
                />
</FlexLayout>
```

6. Add a label next to the ImageButton, with label “Choose a date”.

```
<FlexLayout Direction="Row"
            AlignItems="Center"
            JustifyContent="SpaceEvenly">
        <ImageButton Source="calendaricon.png"
                WidthRequest="50"
                HeightRequest="50"
                IsVisible="true"
                HorizontalOptions="End"
                />
        <Label Text="Choose a date" HorizontalTextAlignment="Start" />
</FlexLayout>
```

7. Your MainPage.xaml should now have the following codes:

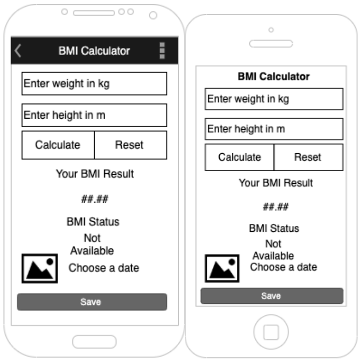


```
<?xml version="1.0" encoding="utf-8" ?>
<ContentPage xmlns="http://xamarin.com/schemas/2014/forms"
             xmlns:x="http://schemas.microsoft.com/winfx/2009/xaml"
             x:Class="BMICalculator.MainPage">

    <StackLayout>

        <StackLayout>
            <Entry x:Name="inputWeight" Placeholder="Enter weight in kg" Keyboard="Numeric"/>
            <Entry x:Name="inputHeight" Placeholder="Enter height in m" Keyboard="Numeric"/>
        </StackLayout>
        <FlexLayout Direction="Row"
                AlignItems="Center"
                JustifyContent="SpaceEvenly">
            <Button x:Name="calculateButton" Text="Calculate" VerticalOptions="CenterAndExpand" WidthRequest="110" BackgroundColor="DarkRed" TextColor="White"  />
            <Button x:Name="resetButton" Text="Reset" VerticalOptions="CenterAndExpand" WidthRequest="110" BackgroundColor="SlateGray" TextColor="White"  />
        </FlexLayout>
        <StackLayout HorizontalOptions="Center">
            <Label Text="Your BMI Result" HorizontalTextAlignment="Center"/>
            <Label x:Name="outputResult" Text="0.00" HorizontalTextAlignment="Center" FontAttributes="Bold"/>
            <Label Text="BMI Status" HorizontalTextAlignment="Center"/>
            <Label x:Name="outputBmiStatus" Text="Not Available" HorizontalTextAlignment="Center" FontAttributes="Bold" />
        </StackLayout>

        <FlexLayout Direction="Row"
                    AlignItems="Center"
                    JustifyContent="SpaceEvenly">
                    <ImageButton Source="calendaricon.png"
                         WidthRequest="50"
                         HeightRequest="50"
                         IsVisible="true"
                         HorizontalOptions="End"
                         />
                    <Label Text="Choose a date" HorizontalTextAlignment="Start" />
        </FlexLayout>

    </StackLayout>

</ContentPage>
```

8. For iOS project **(Windows user can skip steps 8-10),** double click Assets.xcassets.
9. Add new assest. Rename Image with calendaricon.
10. Click and drag the downloaded picture, calendaricon.png into Universal: 1x, 2x, and 3x
11. Run this application. Screen capture the BMI Calculator activity.

Activity 9 – Add a save Button to the apps

1. Refer to the following activities on Android and iOS. In this section, we will add a Button element and layout behind it.



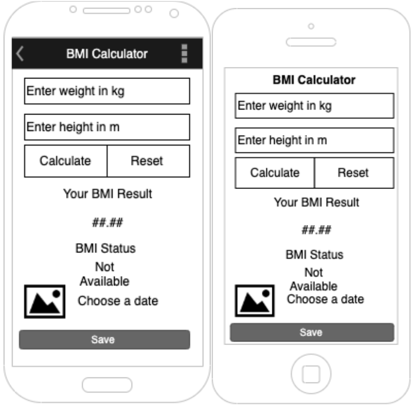
2. Add a StackLayout right after the previous FlexLayout element. Add a button within this StackLayout. Label this button as Save. Set the name of this button as saveButton. Set the width with VerticalOptions as CenterAndExpand.

```
<StackLayout HorizontalOptions="FillAndExpand">
        <Button x:Name="saveButton" Text="Save" VerticalOptions="CenterAndExpand" />
</StackLayout>
```

3. Run this application on Android/iOS emulator. Screen capture the BMI Calculator activity.

Activity 10 – Add a Title to the apps

1. Refer to the following activities on Android and iOS. In this section, we will add a title of this page on top of the screen.



2. In the **App.xaml.cs**, comment the object of MainPage. Add the following code

```
public App()
        {
            InitializeComponent();

            //MainPage = new MainPage();
            MainPage = new NavigationPage(new MainPage());
        }
```

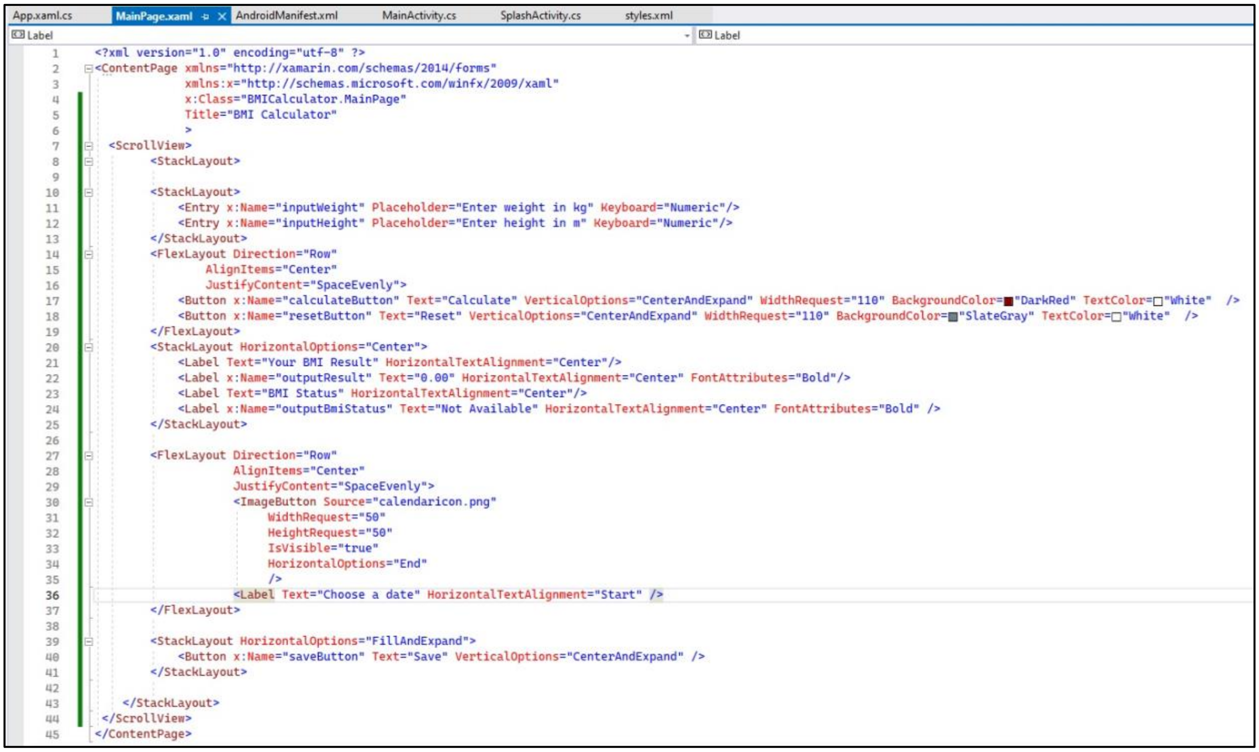
3. Add a title, BMI Calculator. Add the following highlighted code into MainPage.xaml.

```
<ContentPage
   -
   -
   x:Class="BMICalculatorForm.MainPage"
   Title="BMI Calculator"
>
```

4. Add ScrollView to the MainPage.xaml. This should be before the main ScrollView

```
<?xml version="1.0" encoding="utf-8"?>
<ContentPage
    ... >
    <ScrollView>
      <StackLayout>
        …
      </StackLayout>
    </ScrollView>
</ContentPage>
```

5. Your final MainPage.xaml code should have the following codes:

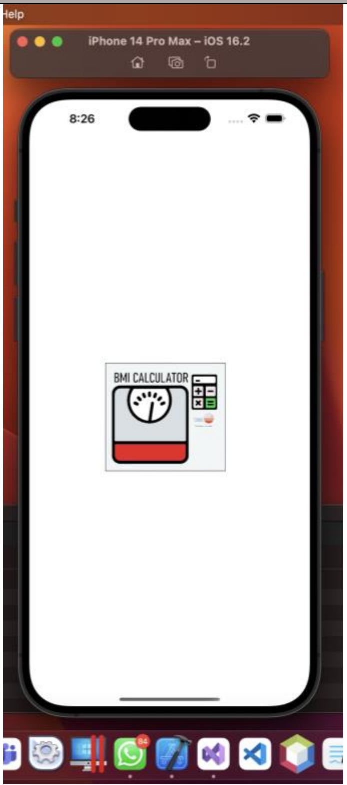


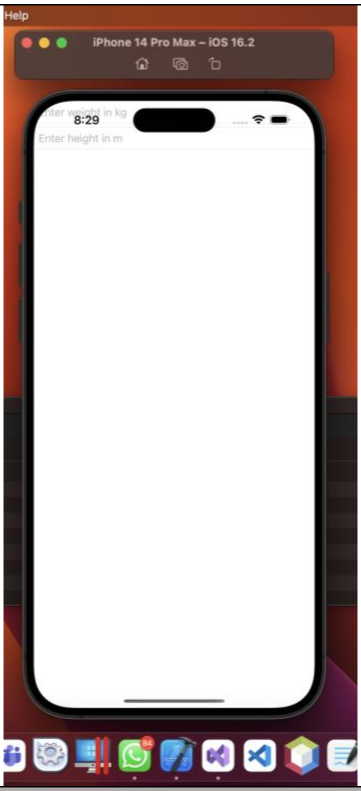
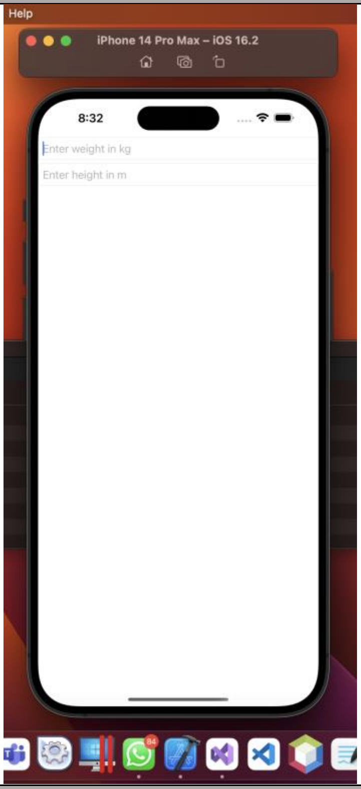
```
<?xml version="1.0" encoding="utf-8" ?>
<ContentPage xmlns="http://xamarin.com/schemas/2014/forms"
             xmlns:x="http://schemas.microsoft.com/winfx/2009/xaml"
             x:Class="BMICalculator.MainPage"
             Title="BMI Calculator"
             >
  <ScrollView>
        <StackLayout>

        <StackLayout>
            <Entry x:Name="inputWeight" Placeholder="Enter weight in kg" Keyboard="Numeric"/>
            <Entry x:Name="inputHeight" Placeholder="Enter height in m" Keyboard="Numeric"/>
        </StackLayout>
        <FlexLayout Direction="Row"
                AlignItems="Center"
                JustifyContent="SpaceEvenly">
            <Button x:Name="calculateButton" Text="Calculate" VerticalOptions="CenterAndExpand" WidthRequest="110" BackgroundColor=■"DarkRed" TextColor=□"White"  />
            <Button x:Name="resetButton" Text="Reset" VerticalOptions="CenterAndExpand" WidthRequest="110" BackgroundColor=■"SlateGray" TextColor=□"White"  />
        </FlexLayout>
        <StackLayout HorizontalOptions="Center">
            <Label Text="Your BMI Result" HorizontalTextAlignment="Center"/>
            <Label x:Name="outputResult" Text="0.00" HorizontalTextAlignment="Center" FontAttributes="Bold"/>
            <Label Text="BMI Status" HorizontalTextAlignment="Center"/>
            <Label x:Name="outputBmiStatus" Text="Not Available" HorizontalTextAlignment="Center" FontAttributes="Bold" />
        </StackLayout>

        <FlexLayout Direction="Row"
                    AlignItems="Center"
                    JustifyContent="SpaceEvenly">
                    <ImageButton Source="calendaricon.png"
                         WidthRequest="50"
                         HeightRequest="50"
                         IsVisible="true"
                         HorizontalOptions="End"
                         />
                    <Label Text="Choose a date" HorizontalTextAlignment="Start" />
        </FlexLayout>

        <StackLayout HorizontalOptions="FillAndExpand">
            <Button x:Name="saveButton" Text="Save" VerticalOptions="CenterAndExpand" />
        </StackLayout>

    </StackLayout>
 </ScrollView>
</ContentPage>
```

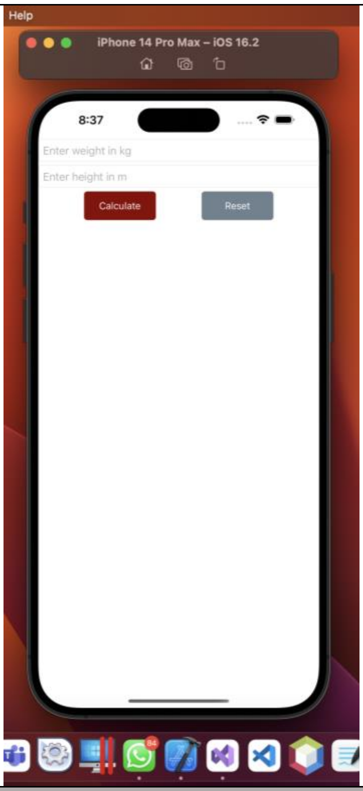
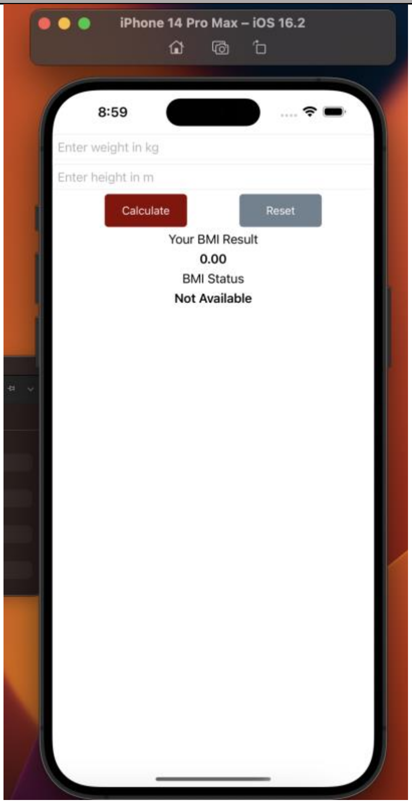
6. Run the application. Screen capture the BMI Calculator activity.

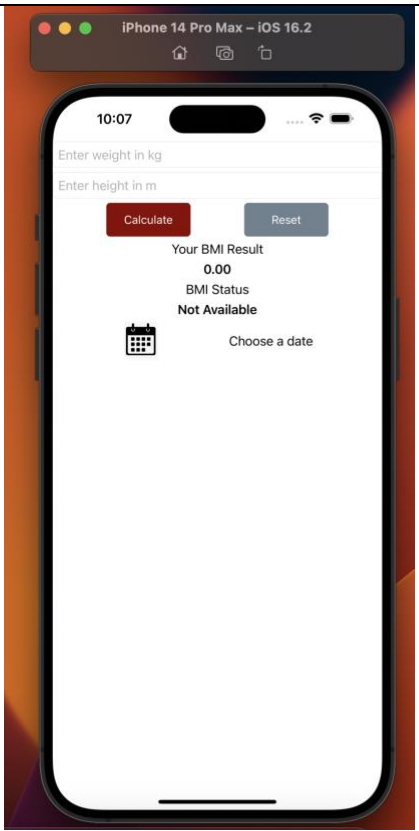
Student ID: SW0107187	Student Name: Akmal Arif Bin Arifin

Instruction: Write your answer in the specified column given.

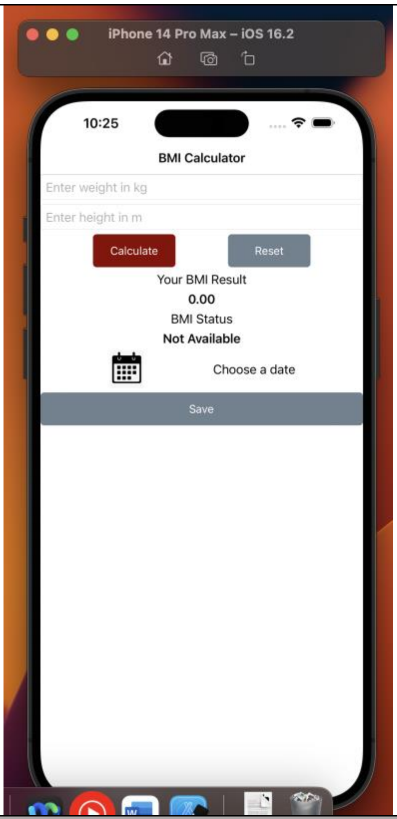
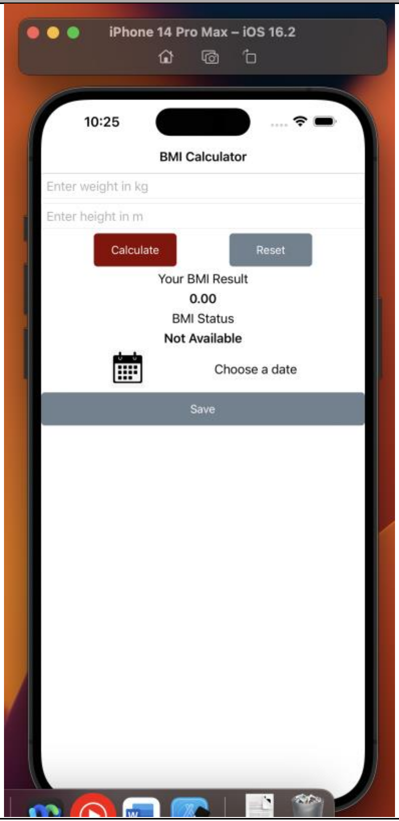
QUESTION	MARKS	ANSWER
Activity 1		
Activity 2		
Screen capture the Splash Screen on Android.	4	 Help iPhone 14 Pro Max – iOS 16.2 8:26 BMI CALCULATOR
Activity 3		
Activity 4		

Run this. Screen capture the BMI Calculator activity.	**2**	 Help iPhone 14 Pro Max – iOS 16.2 8:29 nter weight in kg Enter height in m
Activity 5		
Run this application. Screen capture the BMI Calculator activity.	**2**	 Help iPhone 14 Pro Max – iOS 16.2 8:32 Enter weight in kg Enter height in m
Activity 6		

Run this application. Screen capture the BMI Calculator activity.	**2**	 Help iPhone 14 Pro Max – iOS 16.2 8:37 Enter weight in kg Enter height in m Calculate Reset
Activity 7		
Run this application. Screen capture the BMI Calculator activity.	**3**	 iPhone 14 Pro Max – iOS 16.2 8:59 Enter weight in kg Enter height in m Calculate Reset Your BMI Result 0.00 BMI Status Not Available
Activity 8		

Run this application. Screen capture the BMI Calculator activity.	**3**	 iPhone 14 Pro Max – iOS 16.2 10:07 Enter weight in kg Enter height in m Calculate Reset Your BMI Result 0.00 BMI Status Not Available Choose a date
Activity 9		

Student ID: SW0107187	Student Name: Akmal Arif Bin Arifin

Run this application Screen capture the BMI Calculator activity.	2	 iPhone 14 Pro Max – iOS 16.2 10:25 BMI Calculator Enter weight in kg Enter height in m Calculate Reset Your BMI Result 0.00 BMI Status Not Available Choose a date Save
Activity 10		
Run this application. Screen capture the BMI Calculator activity.	2	 iPhone 14 Pro Max – iOS 16.2 10:25 BMI Calculator Enter weight in kg Enter height in m Calculate Reset Your BMI Result 0.00 BMI Status Not Available Choose a date Save

Student ID: SW0107187	Student Name: Akmal Arif Bin Arifin

TOTAL	**20**	

Rename your word document file to:
Lab 3 <student ID>.docx before the submission.

Submission type: Online