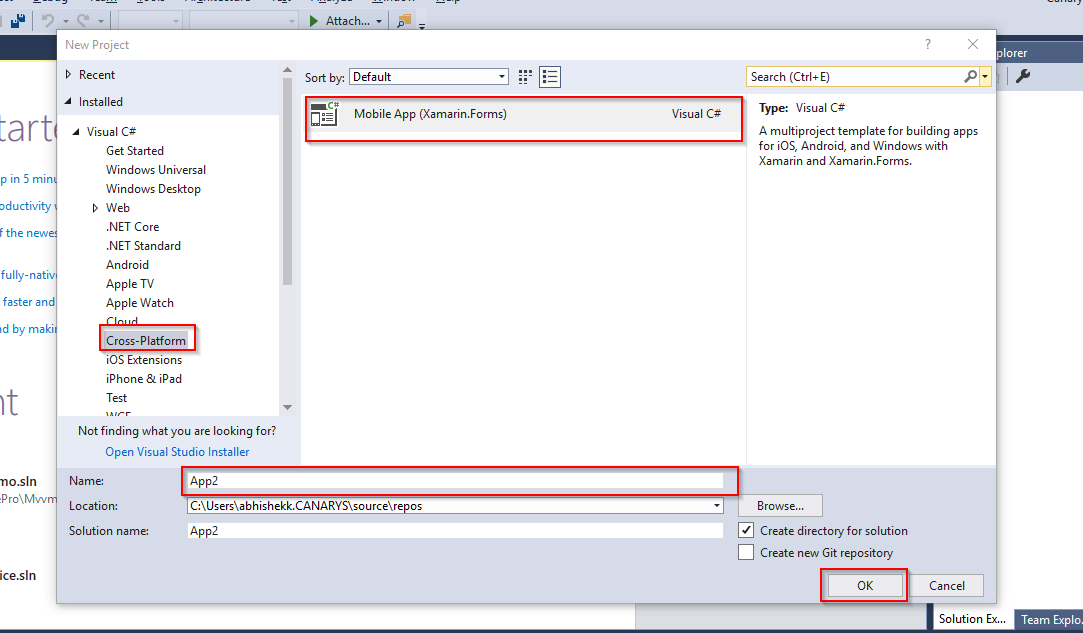
**Xamarin Forms With MVVM**

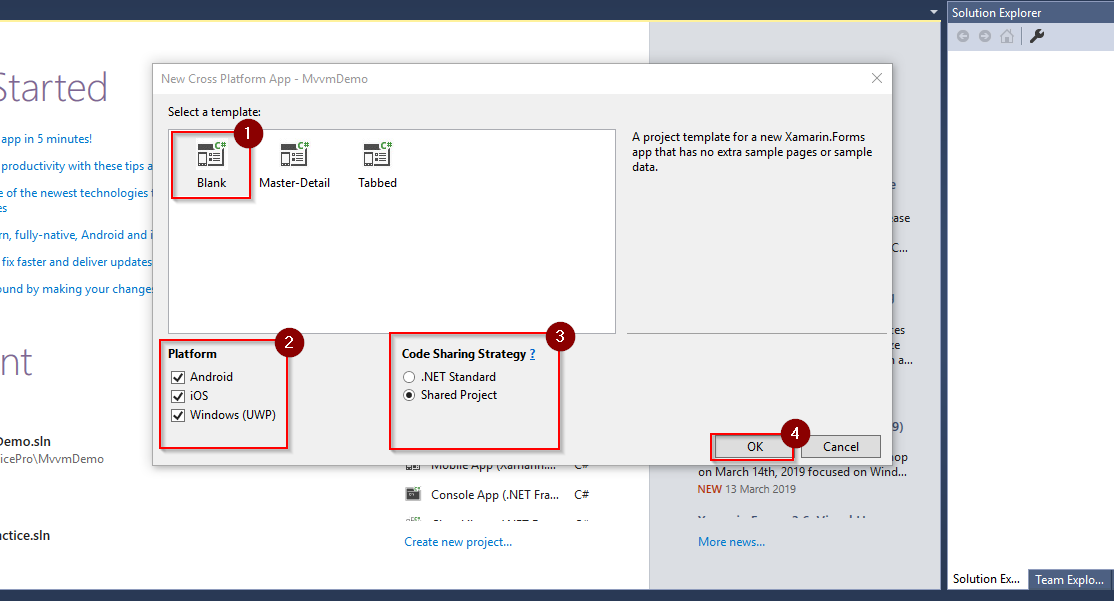
* Open your Visual Studio.
* Go to File🡪 New 🡪 Project.

Create new Xamarin Forms Project. Like I have shown in following image.



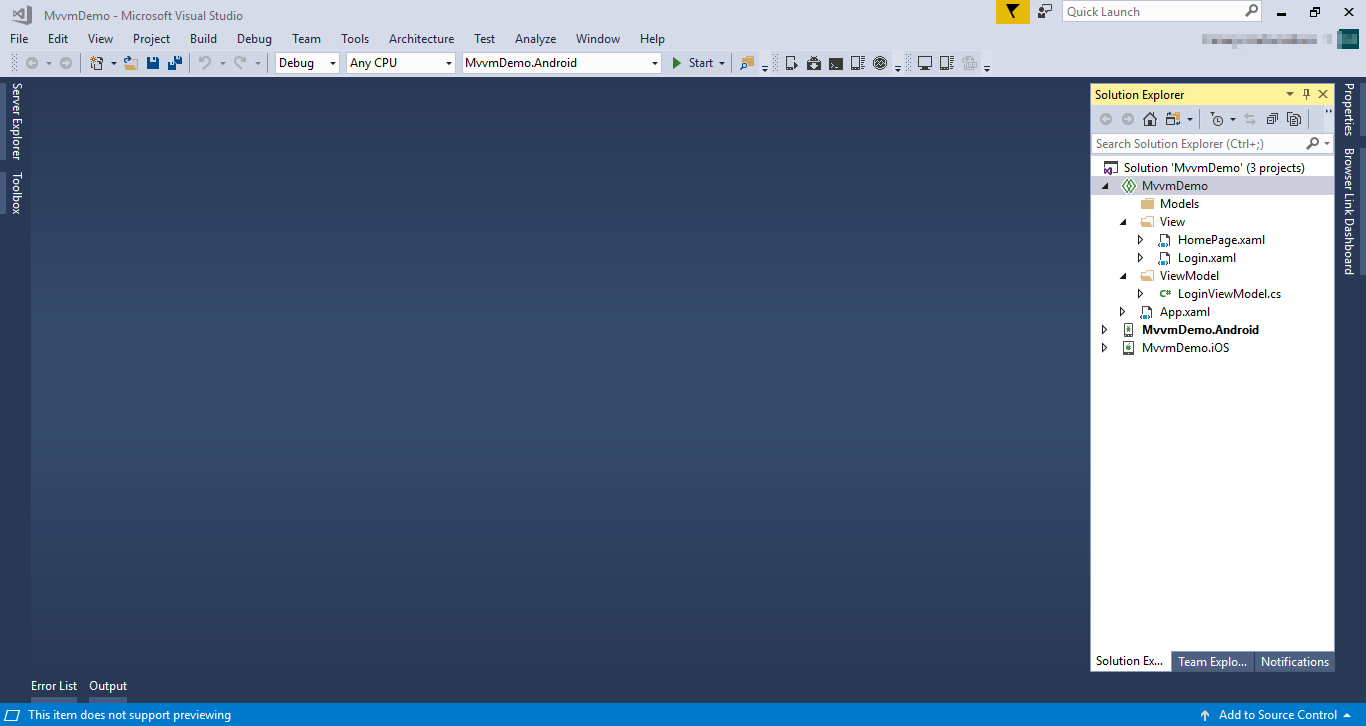
Choose Cross-Platform 🡪 Moblie App 🡪 give project name 🡪 Click on **OK.**

After clicking on OK button you will get the following screen.

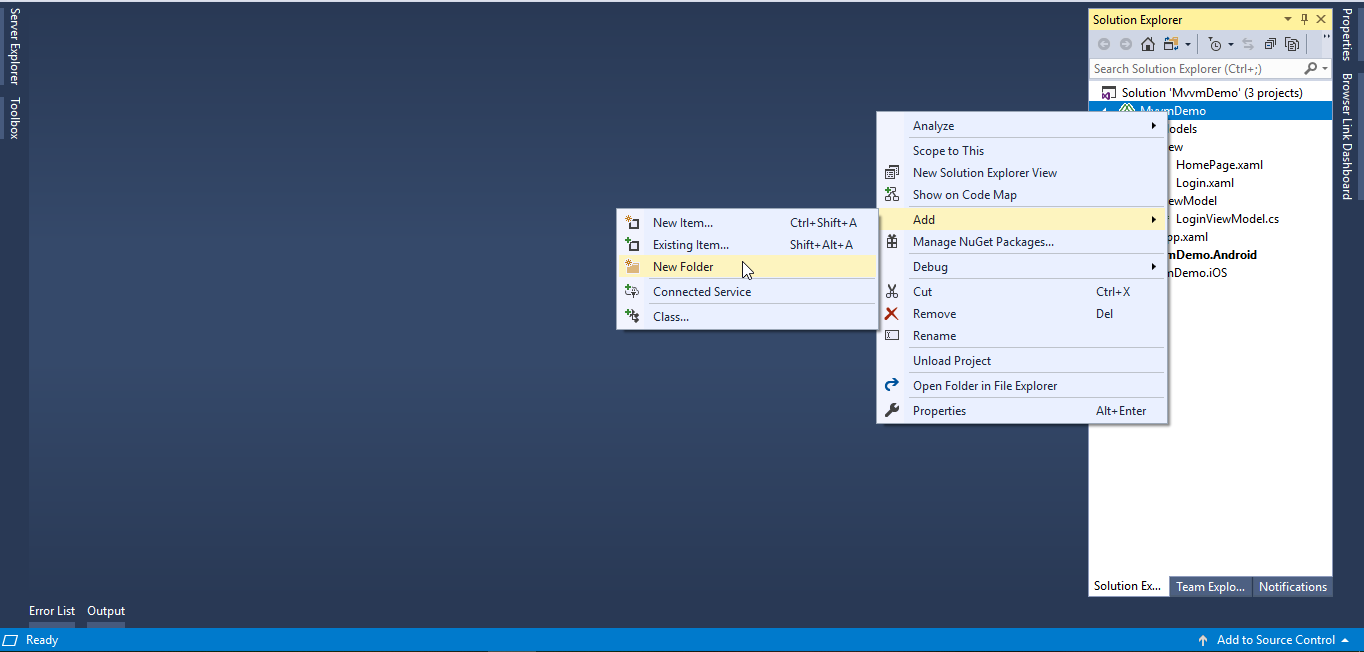


Here you have to follow according to number as I have mention.

After that project will create in your visual studio.it will look like similar to following screen.

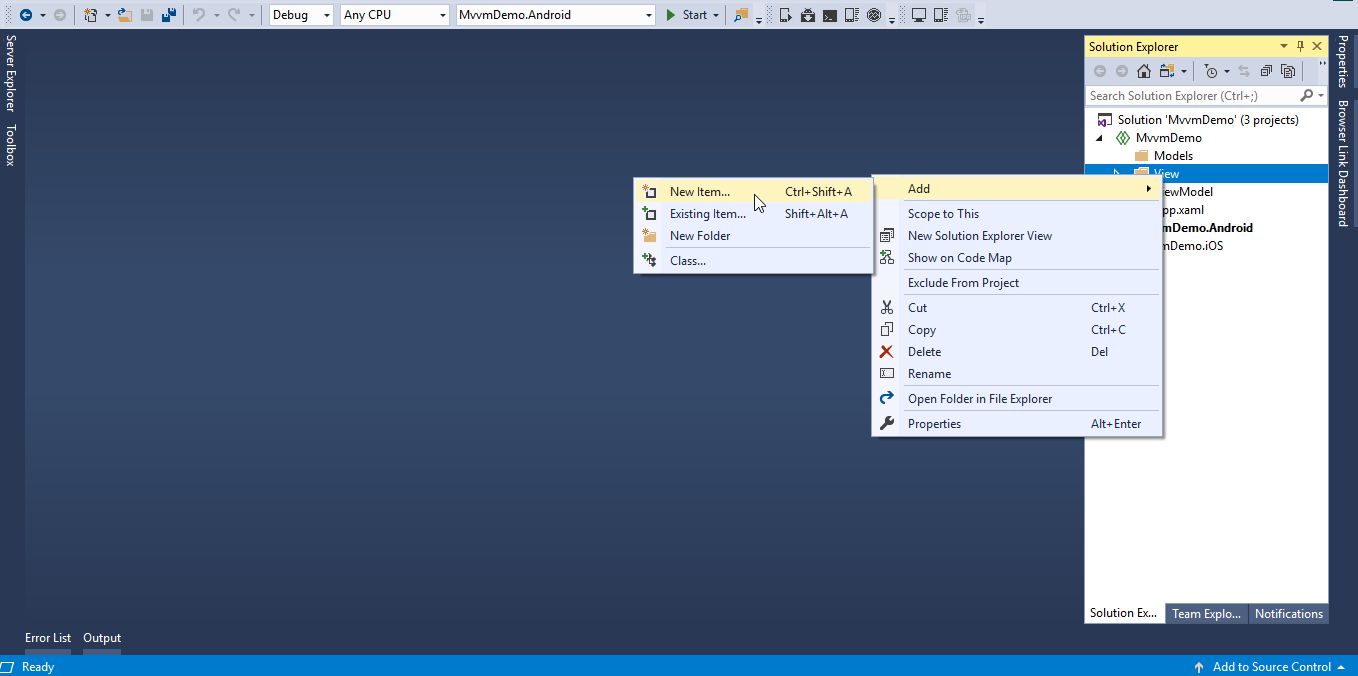


Folder structure would be not here. For that you need to create all three folder.

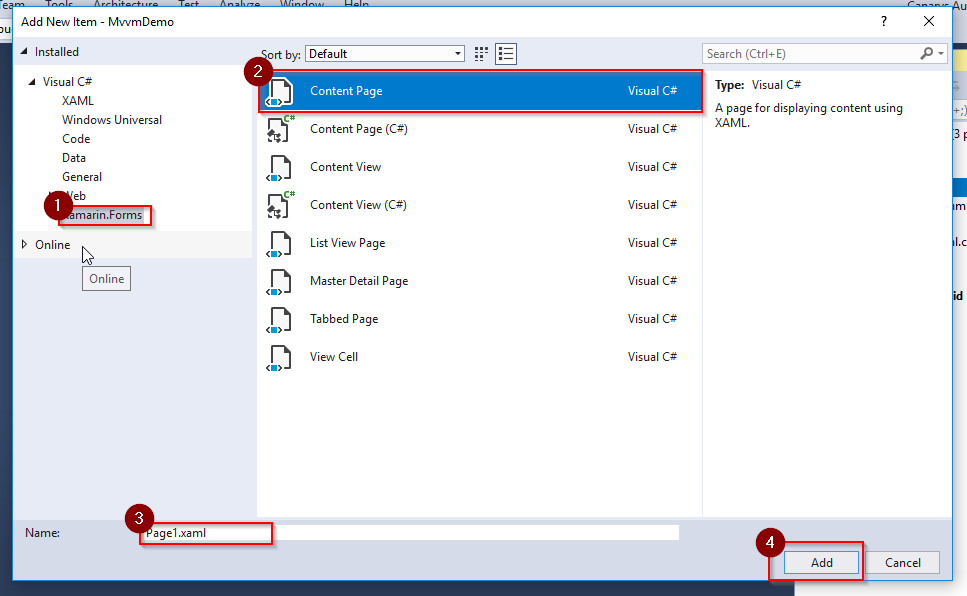


You need to write click on project🡪 Add 🡪 New Folder. Same way you need to create three folder. View,Model,ViewModel.

After that you need to add view in your view folder.

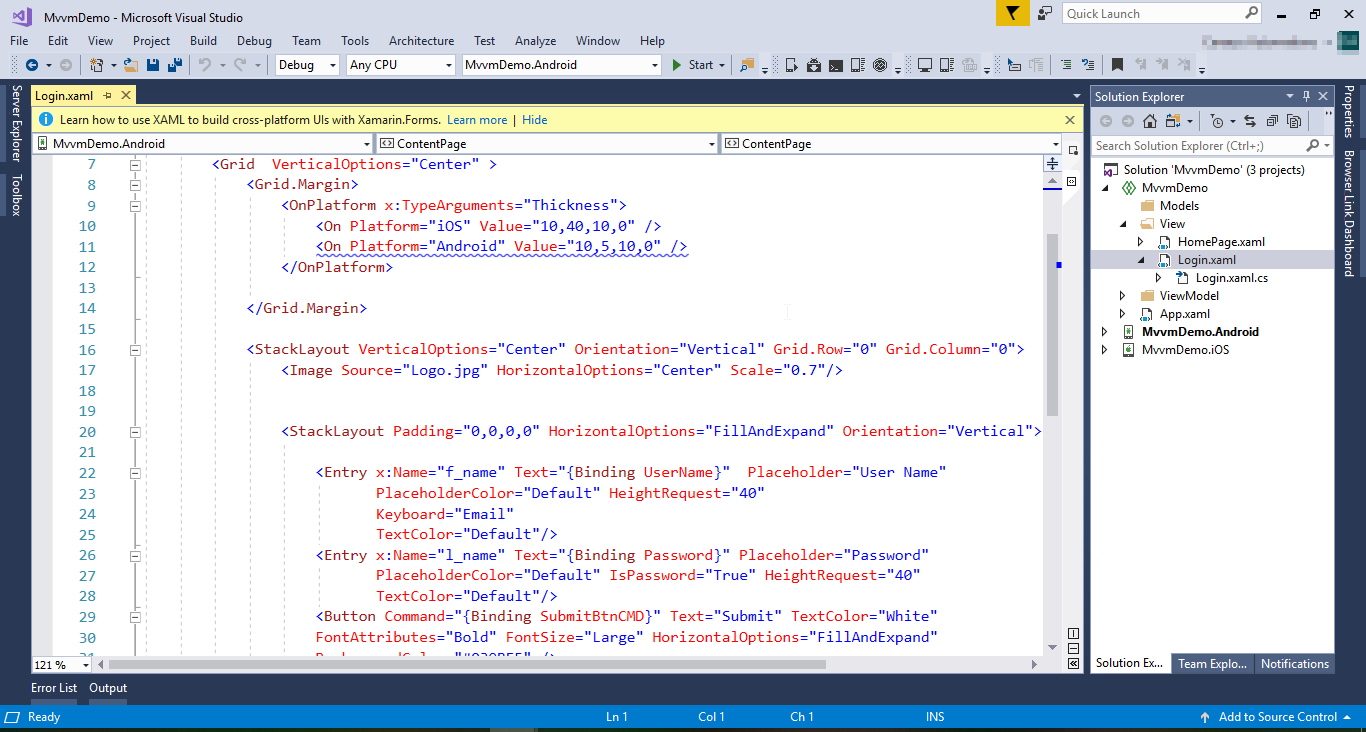


For adding view you need to right click on View🡪Add🡪 New item. Then you will get following screen.



Here you need to follow number wise.

We need to create 2 content page. 1- LoginPage 2-HomePage. So same step you need to follow for creating view.

Now click on LoginPage.xaml and write following design. For reference I have added screen. After that code is available.

<?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="MvvmDemo.View.Login"

BackgroundColor="White">

<ContentPage.Content>

<Grid VerticalOptions="Center" >

<Grid.Margin>

<OnPlatform x:TypeArguments="Thickness">

<On Platform="iOS" Value="10,40,10,0" />

<On Platform="Android" Value="10,5,10,0" />

</OnPlatform>

</Grid.Margin>

<StackLayout VerticalOptions="Center" Orientation="Vertical" Grid.Row="0" Grid.Column="0">

<Image Source="Logo.jpg" HorizontalOptions="Center" Scale="0.7"/>

<StackLayout Padding="0,0,0,0" HorizontalOptions="FillAndExpand" Orientation="Vertical">

<Entry x:Name="f\_name" Text="{Binding UserName}" Placeholder="User Name"

PlaceholderColor="Default" HeightRequest="40"

Keyboard="Email"

TextColor="Default"/>

<Entry x:Name="l\_name" Text="{Binding Password}" Placeholder="Password"

PlaceholderColor="Default" IsPassword="True" HeightRequest="40"

TextColor="Default"/>

<Button Command="{Binding SubmitBtnCMD}" Text="Submit" TextColor="White"

FontAttributes="Bold" FontSize="Large" HorizontalOptions="FillAndExpand"

BackgroundColor="#039BE5" />

</StackLayout>

</StackLayout>

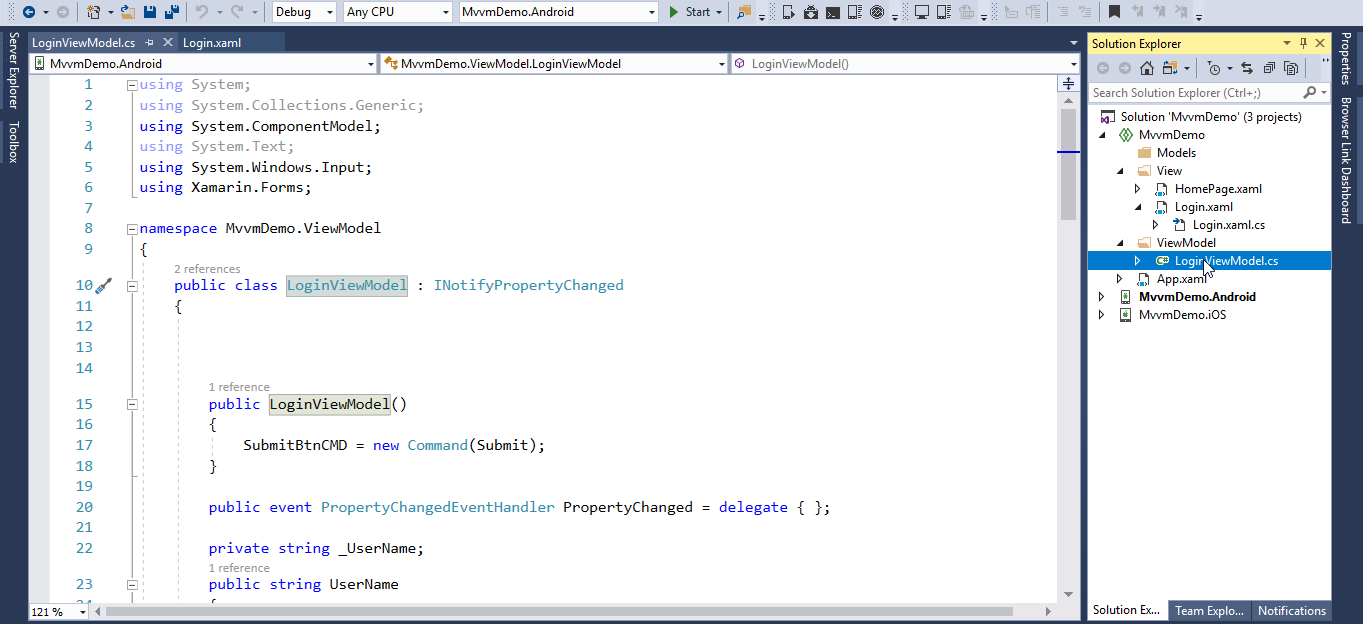
</Grid>

</ContentPage.Content>

</ContentPage>

Just copy this code and paste it in you login page.

**In the next step you need to create one ViewModel**

Same way you need to add one class in you ViewModel folder. Named that class as **LoginViewModel.** For reference I am adding one image code will be available below image.

using System;

using System.Collections.Generic;

using System.ComponentModel;

using System.Text;

using System.Windows.Input;

using Xamarin.Forms;

namespace MvvmDemo.ViewModel

{

public class LoginViewModel : INotifyPropertyChanged

{

public LoginViewModel()

{

SubmitBtnCMD = new Command(Submit);

}

public event PropertyChangedEventHandler PropertyChanged = delegate { };

private string \_UserName;

public string UserName

{

get { return \_UserName; }

set

{

\_UserName = value;

PropertyChanged(this, new PropertyChangedEventArgs("UserName"));

}

}

private string \_password;

public string Password

{

get { return \_password; }

set

{

\_password = value;

PropertyChanged(this, new PropertyChangedEventArgs("Password"));

}

}

public ICommand SubmitBtnCMD { protected set; get; }

public async void Submit()

{

if(UserName =="xamarin" && Password== "xamarin")

{

await Application.Current.MainPage.Navigation.PushAsync(new View.HomePage());

}

else

{

Device.BeginInvokeOnMainThread(async () =>

{

await Application.Current.MainPage.DisplayAlert("Alert", "Login Fail", "Ok");

});

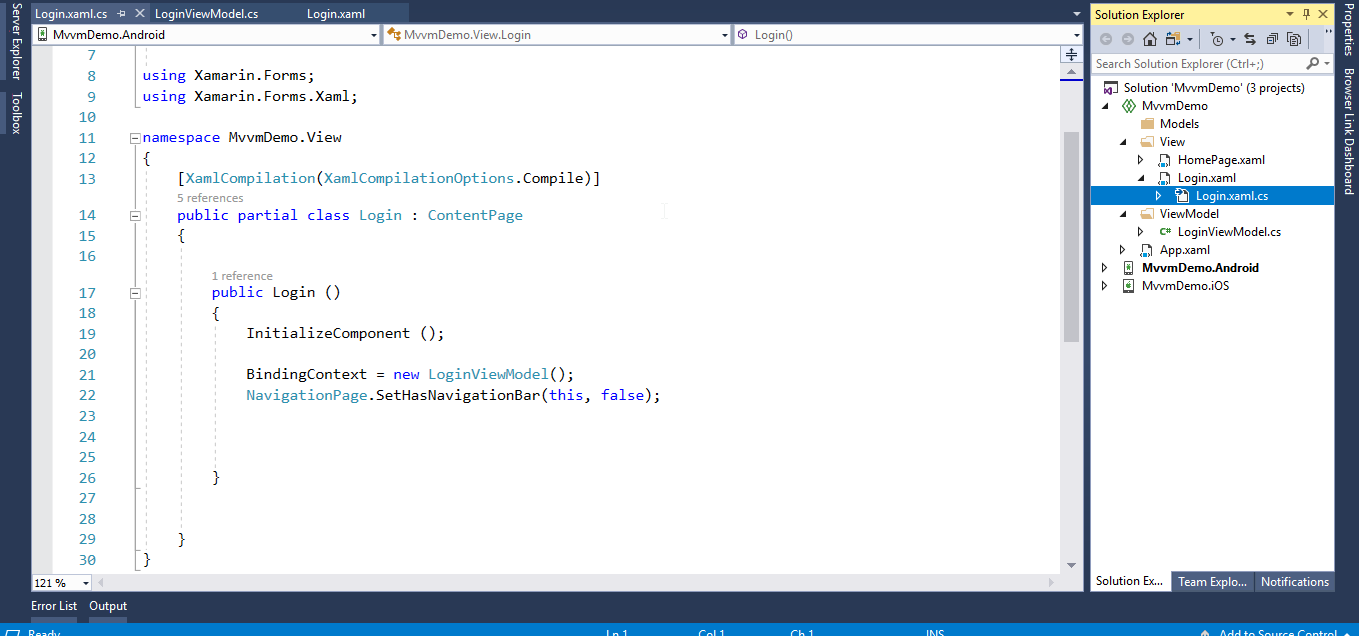
}

}

}

}

Copy this code and paste it in your LoginVewModel.cs File.

Now you need to do Bind your page for that click on Login.xaml.cs file.

Add following code below **InitializeComponent()**.

BindingContext = new LoginViewModel();

NavigationPage.SetHasNavigationBar(this, false);

Now your page binding is also completed.

You can build and deploy your project.

Thank you