University of zakho
Faculty of science
Computer department
Mobile application
Group A

Project about

School Student

Prepared by:

Nergiz Hishyar Ahmad

Israa Sarbast Ali

Kavi Fattah Hussien

Supervisor :

Mr.Yousif Garabet

Table of content:

Chapter 1

1. 1 Introduction

Student Information is used for day-to-day activities in the educational environment. this ptoject, you can learn how to xamarin Forms application using SQLite for Android and Universal Windows Platform with XAML and Visual C# in cross-platform application development.

Now, most of the schools are digitalized. To raise the quality of education many schools turned traditional classed into smart classes. However, communication is something that can't be replaced by anything. So an improved communication is the key to raise education standards. When teachers and parents work collaboratively, students will create automatically the recipe to improve themselves for academic success. In this modern digital education system, we would like to introduce an application, to student school.

1.2 What is Xamarin.Forms?

Xamarin.Forms is an open-source UI framework. Xamarin.Forms allows developers to build Xamarin.Android, Xamarin.iOS, and Windows applications from a single shared codebase.

Xamarin.Forms allows developers to create user interfaces in XAML with code-behind in C#. These interfaces are rendered as performant native controls on each platform.

Xamarin.Forms is for developers with the following goals:

- Share UI layout and design across platforms.
- Share code, test and business logic across platforms.
- Write cross-platform apps in C# with Visual Studio.

Xamarin.Forms provides a consistent API for creating UI elements across platforms. This API can be implemented in either XAML or C# and supports databinding for patterns such as Model-View-ViewModel (MVVM).

At runtime, Xamarin.Forms utilizes platform renderers to convert the cross-platform UI elements into native controls on Xamarin.Android, Xamarin.iOS and UWP. This allows developers to get the native look, feel and performance while realizing the benefits of code sharing across platforms.

Xamarin.Forms applications typically consist of a shared .NET Standard library and individual platform projects. The shared library contains the XAML or C# views and any business logic such as services, models or other code. The platform projects contain any platform-specific logic or packages the application requires.

Chapter 2

MainPage . xaml

```
<?xml version="1.0" encoding="utf-8" ?>
<TabbedPage xmlns="http://xamarin.com/schemas/2014/forms"
             xmlns:x="http://schemas.microsoft.com/winfx/2009/xaml"
             x:Class="School.MainPage"
            xmlns:android="clr-
namespace:Xamarin.Forms.PlatformConfiguration.AndroidSpecific;assembly=
Xamarin.Forms.Core"
            android:TabbedPage.ToolbarPlacement="Bottom"
            xmlns:local="clr-namespace:School;assembly=School"
            BarBackgroundColor="#2196F3"
            SelectedTabColor="#FFD600"
            BarTextColor="White"
            >

    <local:ST Title="Students" IconImageSource="student.png" />
    <local:T Title="Teachers" IconImageSource="teacher.png"/>


</TabbedPage>
```

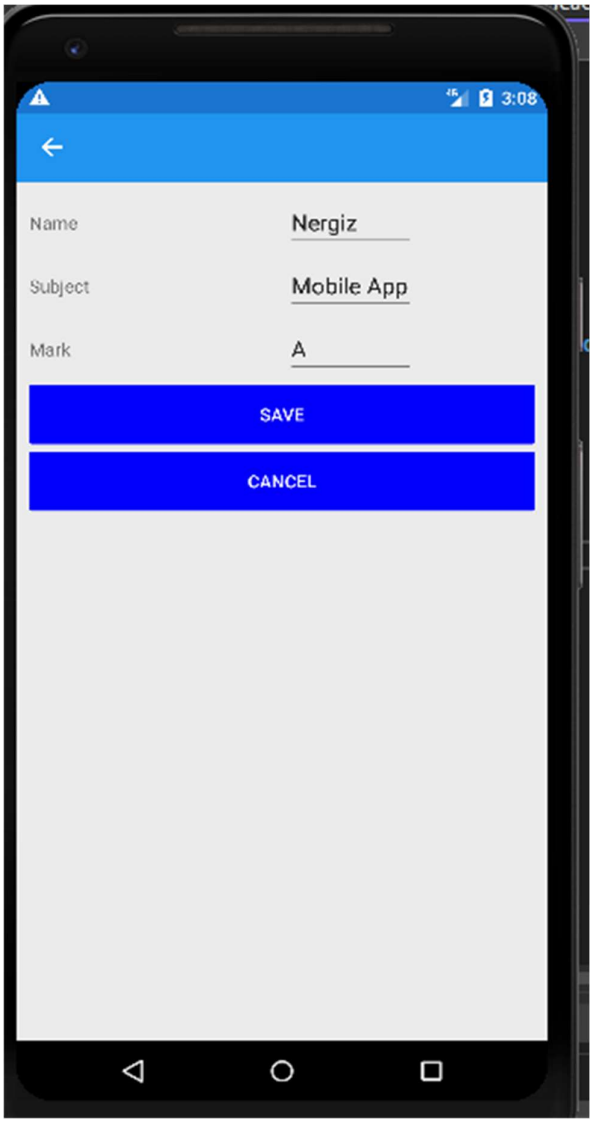
App.xaml.cs

```
using System;
using Xamarin.Forms;
using Xamarin.Forms.Xaml;

namespace School
{
    public partial class App : Application
    {
        static StudDB database;
        static TDB databaseT;
        public App()
        {
            InitializeComponent();

            MainPage = new NavigationPage(new School.MainPage());
        }

        public static StudDB Database
        {
            get
            {
                if (database == null)
                {
                    database = new
StudDB(DependencyService.Get<IStdLocHelper>().GetLocalFilePath("Student1.db3"));
                }
                return database;
            }
        }

        public static TDB DatabaseT
        {
            get
            {
                if (databaseT == null)
                {
                    databaseT = new
TDB(DependencyService.Get<IStdLocHelper>().GetLocalFilePath("Student1.db3"));
                }
                return databaseT;
            }
        }

        protected override void OnStart()
        {
        }

        protected override void OnSleep()
        {
        }

        protected override void OnResume()
        {
        }
    }
}
```

AddStudent.xaml :

```
?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="School.AddStudent">
    <ContentPage.Content>
        <StackLayout Padding="10" HorizontalOptions="FillAndExpand"
VerticalOptions="FillAndExpand">
            <Grid>
                <Label Text="Name" Grid.Row="0" Grid.Column="0"
HorizontalOptions="Start" WidthRequest="100" VerticalOptions="Center"/>
                <Entry Text="{Binding stdname}" Grid.Row="0"
Grid.Column="1" HorizontalOptions="Start" WidthRequest="100"
VerticalOptions="Center"/>
                <Label Text="Subject" Grid.Row="1" Grid.Column="0"
HorizontalOptions="Start" WidthRequest="100" VerticalOptions="Center"/>
                <Entry Text="{Binding stdsubj}" Grid.Row="1"
Grid.Column="1" HorizontalOptions="Start" WidthRequest="100"
VerticalOptions="Center"/>
                <Label Text="Mark" Grid.Row="2" Grid.Column="0"
HorizontalOptions="Start" WidthRequest="100" VerticalOptions="Center"/>
                <Entry Text="{Binding subjmark}" Grid.Row="2"
Grid.Column="1" HorizontalOptions="Start" WidthRequest="100"
VerticalOptions="Center"/>
            </Grid>
            <Button Text="Save" HorizontalOptions="FillAndExpand"
BackgroundColor="Blue" TextColor="White"  Clicked="Save_Clicked"/>
            <Button Text="Cancel" HorizontalOptions="FillAndExpand"
BackgroundColor="Blue" TextColor="White" Clicked="Cancel_Clicked"/>
        </StackLayout>
    </ContentPage.Content>
</ContentPage>
```

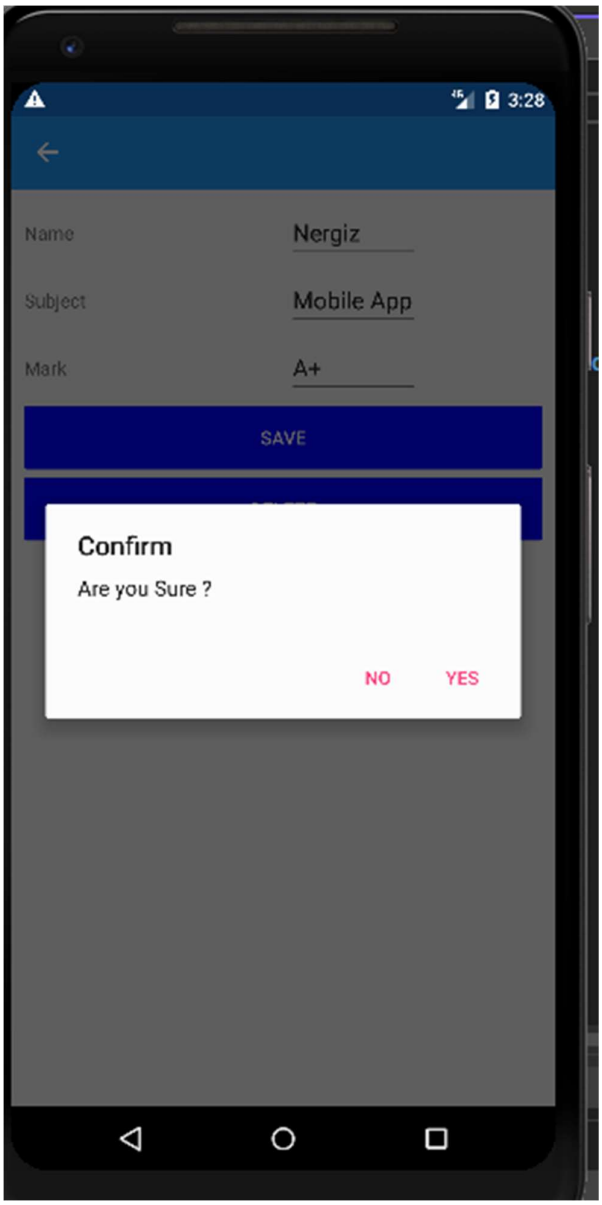
Add student's information:



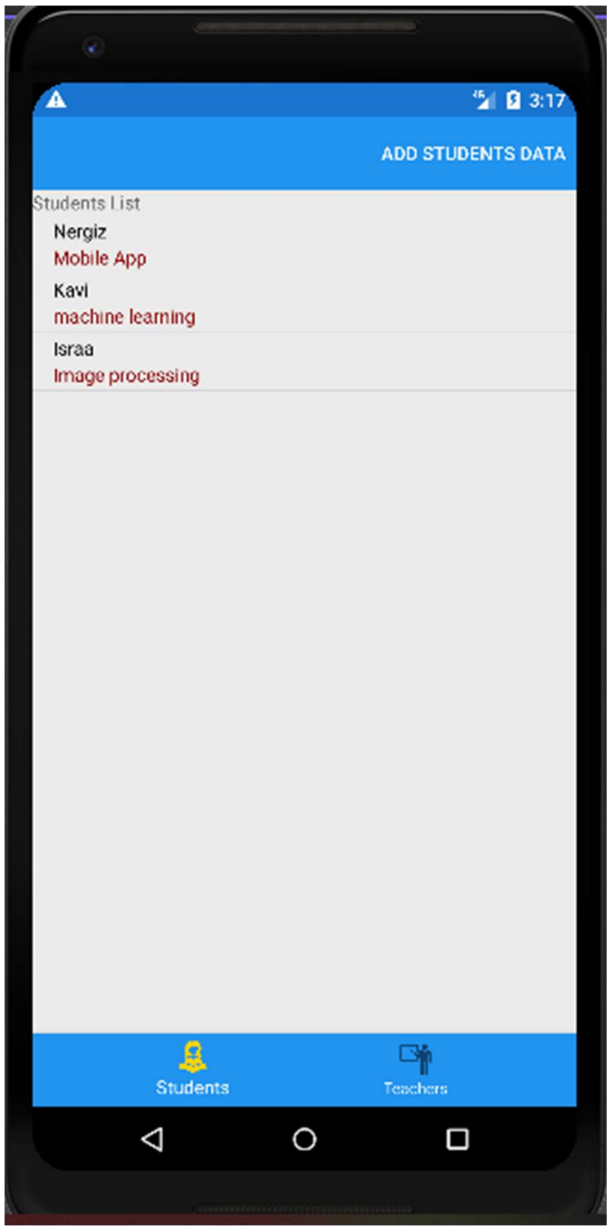
EditStudent . xaml:

```
<?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="School.EditStudent">
    <ContentPage.Content>
        <StackLayout Padding="10" HorizontalOptions="FillAndExpand"
VerticalOptions="FillAndExpand">
            <Grid>
                <Label Text="Name" Grid.Row="0" Grid.Column="0"
HorizontalOptions="Start" WidthRequest="100" VerticalOptions="Center"/>
                <Entry Text="{Binding stdname}" Grid.Row="0"
Grid.Column="1" HorizontalOptions="Start" WidthRequest="100"
VerticalOptions="Center"/>
                <Label Text="Subject" Grid.Row="1" Grid.Column="0"
HorizontalOptions="Start" WidthRequest="100" VerticalOptions="Center"/>
                <Entry Text="{Binding stdsubj}" Grid.Row="1"
Grid.Column="1" HorizontalOptions="Start" WidthRequest="100"
VerticalOptions="Center"/>
                <Label Text="Mark" Grid.Row="2" Grid.Column="0"
HorizontalOptions="Start" WidthRequest="100" VerticalOptions="Center"/>
                <Entry Text="{Binding subjmark}" Grid.Row="2"
Grid.Column="1" HorizontalOptions="Start" WidthRequest="100"
VerticalOptions="Center"/>
            </Grid>
            <Button Text="Save" HorizontalOptions="FillAndExpand"
BackgroundColor="Blue" TextColor="White"  Clicked="Save_Clicked"/>
            <Button Text="Delete" HorizontalOptions="FillAndExpand"
BackgroundColor="Blue" TextColor="White" Clicked="OnDeleteClicked"/>
        </StackLayout>
    </ContentPage.Content>

</ContentPage>
```

Delete student’s informations:



View list students:



ST.xaml:

```
<ContentPage xmlns="http://xamarin.com/schemas/2014/forms"
             xmlns:x="http://schemas.microsoft.com/winfx/2009/xaml"
             x:Class="School.ST">

    <ContentPage.Content>
        <ListView x:Name="StudListview"
ItemSelected="Student_Itemselected" Header="Students List">
            <ListView.ItemTemplate>
                <DataTemplate>
                    <TextCell Text="{Binding stdname}" Detail="{Binding
stdsubj}" DetailColor="DarkRed" TextColor="Black"/>
                </DataTemplate>
            </ListView.ItemTemplate>
        </ListView>
    </ContentPage.Content>

          </ContentPage>
```

Student.cs

```
using System;
using System.Collections.Generic;
using System.Text;
using SQLite;

namespace School
{
    public class Student
    {
        [PrimaryKey, AutoIncrement]
        public int stdid { get; set; }
        [NotNull]
        public string stdname { get; set; }
        public string stdsubj { get; set; }
        public int subjmark { get; set; }
    }
}
```

T.xaml

```
<ContentPage xmlns="http://xamarin.com/schemas/2014/forms"

xmlns:x="http://schemas.microsoft.com/winfx/2009/xaml"
             x:Class="School.T"
             >

    <ContentPage.Content>
        <ListView x:Name="TeacherListview"
ItemSelected="Teacher_Itemselected" Header="Teachers List">
            <ListView.ItemTemplate>
                <DataTemplate>
                    <TextCell Text="{Binding tname}"
Detail="{Binding tsubj}" DetailColor="DarkRed"
TextColor="Black" />
                </DataTemplate>
            </ListView.ItemTemplate>
        </ListView>
    </ContentPage.Content>

        <ContentPage/>
```

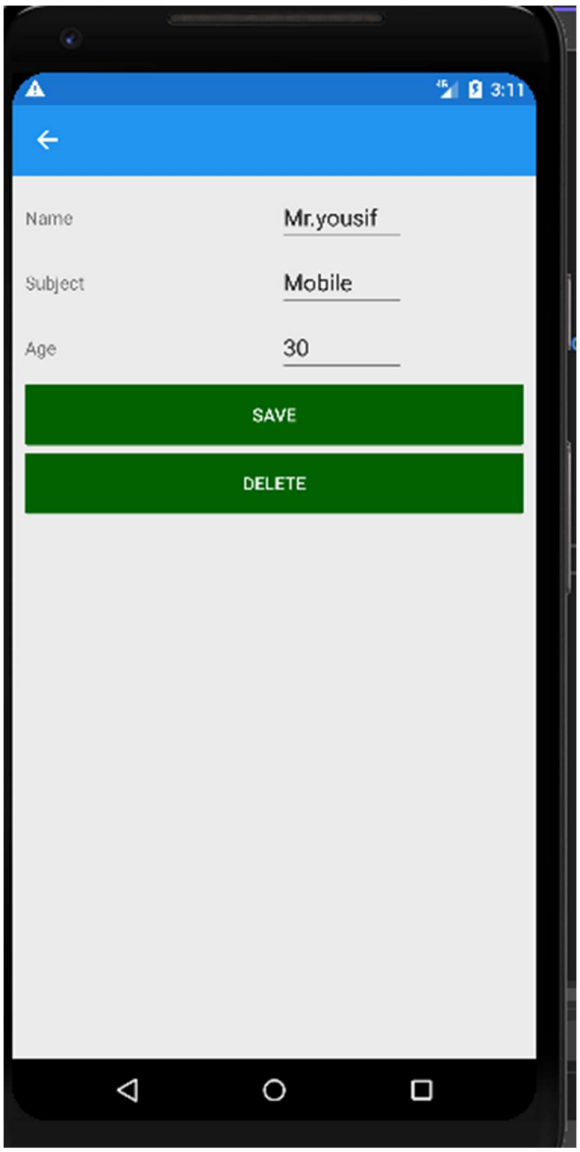
T.xaml.cs

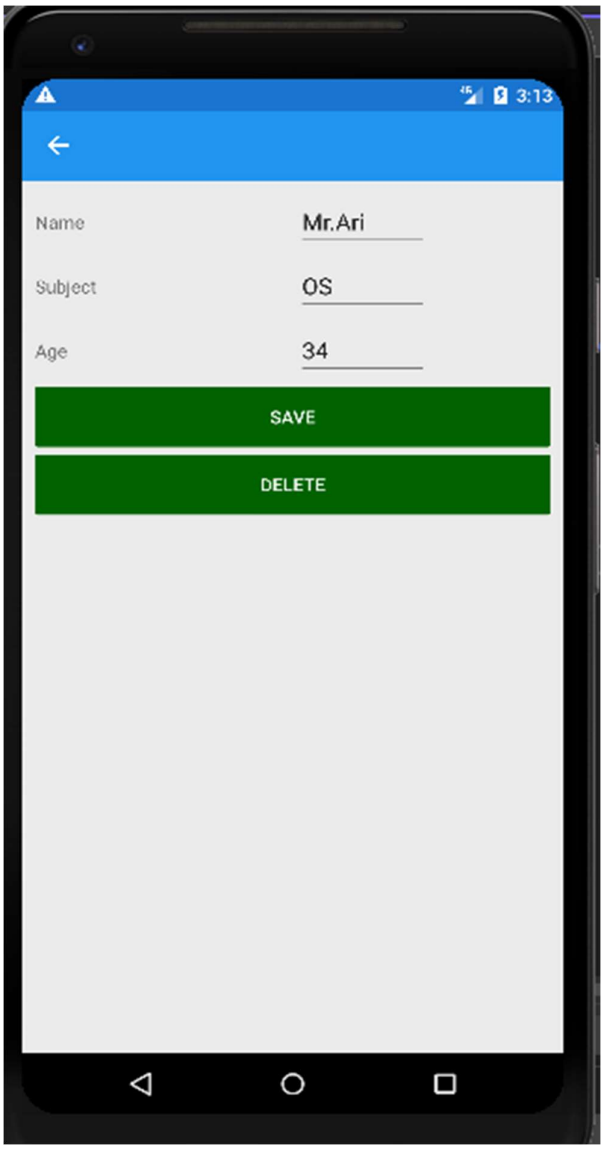
```
using System;
using System.Collections.Generic;
using System.Linq;
using System.Text;
using System.Threading.Tasks;

using Xamarin.Forms;
using Xamarin.Forms.Xaml;

namespace School
{
    [XamlCompilation(XamlCompilationOptions.Compile)]
    public partial class T : ContentPage
    {
        public T()
        {
            InitializeComponent();
            this.Title = "Student List";
            var toolbarItem = new ToolbarItem { Text = "Add Teachers
Data" };
            toolbarItem.Clicked += async (sender, e) =>
            {
                await Navigation.PushAsync(new AddTeacher() {
BindingContext = new Teachers() });
            };
            ToolbarItems.Add(toolbarItem);
        }
        protected async override void OnAppearing()
        {
            base.OnAppearing();
            TeacherListview.ItemsSource = await
App.DatabaseT.GetTeachersAsync();
        }
        async void Teacher_Itemselected(object sender,
SelectedItemChangedEventArgs e)
        {
            if (e.SelectedItem != null)
            {
                await Navigation.PushAsync(new EditTeacher() {
BindingContext = e.SelectedItem as Teachers });
            }
        }
    }
        }
```

AddTeacher.xaml

```
<ContentPage xmlns="http://xamarin.com/schemas/2014/forms"
             xmlns:x="http://schemas.microsoft.com/winfx/2009/xaml"
             x:Class="School.AddTeacher">
    <ContentPage.Content>
        <StackLayout Padding="10" HorizontalOptions="FillAndExpand"
VerticalOptions="FillAndExpand">
            <Grid>
                <Label Text="Name" Grid.Row="0" Grid.Column="0"
HorizontalOptions="Start" WidthRequest="100" VerticalOptions="Center"/>
                <Entry Text="{Binding tname}" Grid.Row="0"
Grid.Column="1" HorizontalOptions="Start" WidthRequest="100"
VerticalOptions="Center"/>
                <Label Text="Subject" Grid.Row="1" Grid.Column="0"
HorizontalOptions="Start" WidthRequest="100" VerticalOptions="Center"/>
                <Entry Text="{Binding tsubj}" Grid.Row="1"
Grid.Column="1" HorizontalOptions="Start" WidthRequest="100"
VerticalOptions="Center"/>
                <Label Text="Age" Grid.Row="2" Grid.Column="0"
HorizontalOptions="Start" WidthRequest="100" VerticalOptions="Center"/>
                <Entry Text="{Binding tage}" Grid.Row="2"
Grid.Column="1" HorizontalOptions="Start" WidthRequest="100"
VerticalOptions="Center"/>
            </Grid>
            <Button Text="Save" HorizontalOptions="FillAndExpand"
BackgroundColor="DarkGreen" TextColor="White"  Clicked="Save_Clicked"/>
            <Button Text="Cancel" HorizontalOptions="FillAndExpand"
BackgroundColor="DarkGreen" TextColor="White"
Clicked="Cancel_Clicked"/>
        </StackLayout>
    </ContentPage.Content>
</ContentPage>
```

Add teacher’s informations:

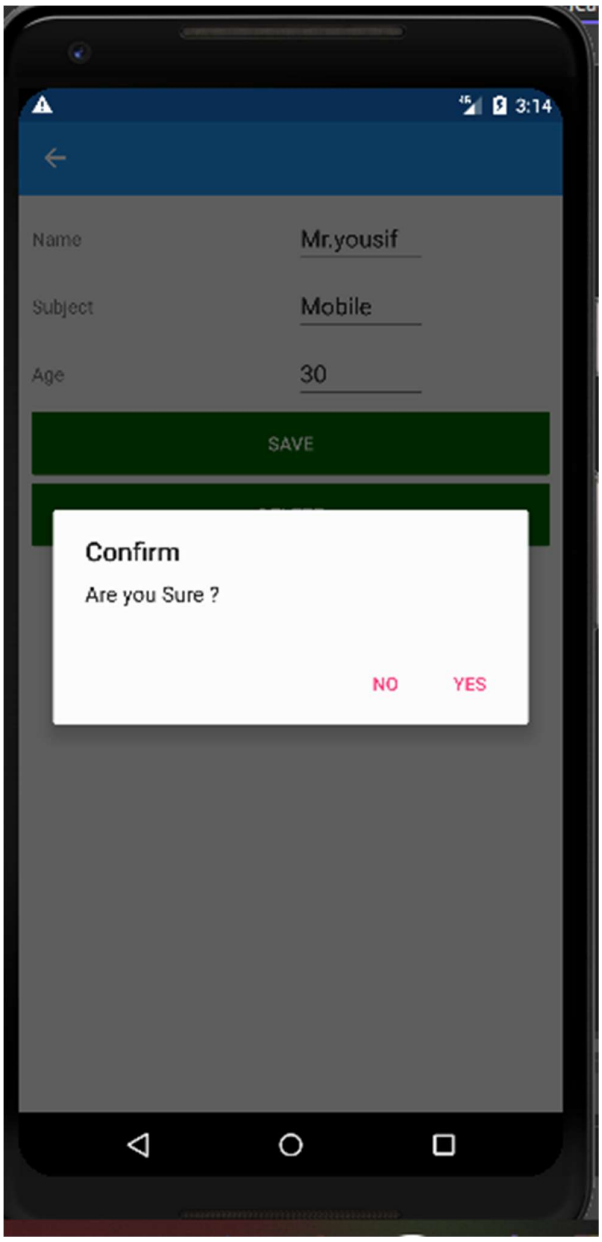




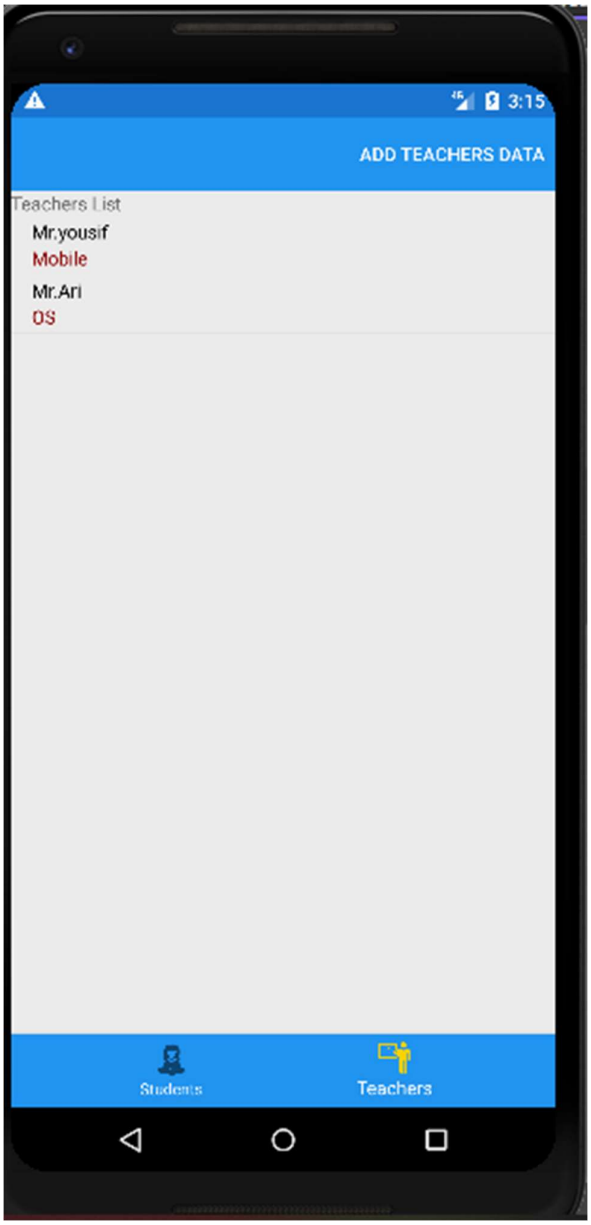
EditTeacher.xaml

```
<?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="School.EditTeacher">
    <ContentPage.Content>
        <StackLayout Padding="10" HorizontalOptions="FillAndExpand"
VerticalOptions="FillAndExpand">
            <Grid>
                <Label Text="Name" Grid.Row="0" Grid.Column="0"
HorizontalOptions="Start" WidthRequest="100" VerticalOptions="Center"/>
                <Entry Text="{Binding tname}" Grid.Row="0"
Grid.Column="1" HorizontalOptions="Start" WidthRequest="100"
VerticalOptions="Center"/>
                <Label Text="Subject" Grid.Row="1" Grid.Column="0"
HorizontalOptions="Start" WidthRequest="100" VerticalOptions="Center"/>
                <Entry Text="{Binding tsubj}" Grid.Row="1"
Grid.Column="1" HorizontalOptions="Start" WidthRequest="100"
VerticalOptions="Center"/>
                <Label Text="Age" Grid.Row="2" Grid.Column="0"
HorizontalOptions="Start" WidthRequest="100" VerticalOptions="Center"/>
                <Entry Text="{Binding tage}" Grid.Row="2"
Grid.Column="1" HorizontalOptions="Start" WidthRequest="100"
VerticalOptions="Center"/>
            </Grid>
            <Button Text="Save" HorizontalOptions="FillAndExpand"
BackgroundColor="DarkGreen" TextColor="White"  Clicked="Save_Clicked"/>
            <Button Text="Delete" HorizontalOptions="FillAndExpand"
BackgroundColor="DarkGreen" TextColor="White"
Clicked="OnDeleteClicked"/>
        </StackLayout>
    </ContentPage.Content>

        </ContentPage>
```

Delete teacher’s informations:



View list students:



Chapter 3

Conclusion

The most important functions that School Application offers are solutions to managing data transactions efficiently and simplifying your Schoolwork's operations. The Application should be able to meet your Schoolwork's requirements and have the capacity to handle your workload. Train your Staff and teachers adequately on its functionality and make sure they understand how it works to avoid errors. Get feedback from them on whether it is user-friendly and fast enough for your schoolwork. School Application is one of the leading Applications on phone It allows quick, clear and detailed overview of all the components of running a successful in Schoolwork . Its user-friendly interface, numerous functions, easy access, and offered detailed information provide the entire School Staff with the ability to perform their work in the most effective way.

References

- https://schoolplusapp.com/mobile-app-for-schools.html
- https://docs.microsoft.com/en-us/xamarin/get-started/what-is-xamarin-forms