**The laboratory work 4**

**Navigation in a Multi-Page Xamarin.Forms Application**

## **Update the app with Visual Studio**

1. Launch Visual Studio. In the start window, click the **Notes** solution in the recent projects/solutions list, or click **Open a project or solution**, and in the **Open Project/Solution** dialog select the solution file for the Notes project:



1. In **Solution Explorer**, right-click on the **Notes** project and select **Add > New Folder**:



1. In **Solution Explorer**, name the new folder **Models**:



1. In **Solution Explorer**, select the **Models** folder, right-click, and select **Add > New Item...**:



1. In the **Add New Item** dialog, select **Visual C# Items > Class**, name the new file **Note**, and click the **Add** button:



This will add a class named **Note** to the **Models** folder of the **Notes** project.

1. In **Note.cs**, remove all of the template code and replace it with the following code:

using System;

namespace Notes.Models

{

 public class Note

 {

 public string Filename {get; set;}

 public string Text {get; set;}

 public DateTime Date {get; set;}

 }

}

* This class defines a Note model that will store data about each note in the application.

Save the changes to **Note.cs** by pressing **CTRL+S**, and close the file.

1. In **Solution Explorer**, right-click on the **Notes** project and select **Add > New Item..**. In the **Add New Item** dialog, select **Visual C# Items > Xamarin.Forms > Content Page**, name the new file **NoteEntryPage**, and click the **Add** button:



This will add a new page named **NoteEntryPage** to the root folder of the project. This page will be the second page in the application.

1. In **NoteEntryPage.xaml**, remove all of the template code and replace it with 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="Notes.NoteEntryPage" Title="Note Entry"> <StackLayout Margin="20"> <Editor Placeholder="Enter your note" Text="{Binding Text}" HeightRequest="100" /> <Grid> <Grid.ColumnDefinitions> <ColumnDefinition Width="\*" /> <ColumnDefinition Width="\*" /> </Grid.ColumnDefinitions> <Button Text="Save" Clicked="OnSaveButtonClicked" /> <Button Grid.Column="1" Text="Delete" Clicked="OnDeleteButtonClicked"/> </Grid> </StackLayout></ContentPage> |

This code declaratively defines the user interface for the page, which consists of an [Editor](https://docs.microsoft.com/en-us/dotnet/api/xamarin.forms.editor) for text input, and two [Button](https://docs.microsoft.com/en-us/dotnet/api/xamarin.forms.button) instances that direct the application to save or delete a file. The two Button instances are horizontally laid out in a [Grid](https://docs.microsoft.com/en-us/dotnet/api/xamarin.forms.grid), with the Editor and Grid being vertically laid out in a [StackLayout](https://docs.microsoft.com/en-us/dotnet/api/xamarin.forms.stacklayout). In addition, the Editor uses data binding to bind to the Text property of the Note model. For more information about data binding, see [Data binding](https://docs.microsoft.com/en-us/xamarin/get-started/quickstarts/deepdive#data-binding) in the [Xamarin.Forms Quickstart Deep Dive](https://docs.microsoft.com/en-us/xamarin/get-started/quickstarts/deepdive).

Save the changes to **NoteEntryPage.xaml** by pressing **CTRL+S**, and close the file.

1. In **NoteEntryPage.xaml.cs**, remove all of the template code and replace it with the following code:

|  |
| --- |
| using System;using System.IO;using Xamarin.Forms;using Notes.Models;namespace Notes{ public partial class NoteEntryPage : ContentPage { public NoteEntryPage() { InitializeComponent(); } async void OnSaveButtonClicked(object sender, EventArgs e) { var note = (Note)BindingContext; if (string.IsNullOrWhiteSpace(note.Filename)) { // Save var filename = Path.Combine(App.FolderPath, $"{Path.GetRandomFileName()}.notes.txt"); File.WriteAllText(filename, note.Text); } else { // Update File.WriteAllText(note.Filename, note.Text); } await Navigation.PopAsync(); } async void OnDeleteButtonClicked(object sender, EventArgs e) { var note = (Note)BindingContext; if (File.Exists(note.Filename)) { File.Delete(note.Filename); } await Navigation.PopAsync(); } }} |

This code stores a Note instance, which represents a single note, in the [BindingContext](https://docs.microsoft.com/en-us/dotnet/api/xamarin.forms.bindableobject.bindingcontext#Xamarin_Forms_BindableObject_BindingContext) of the page. When the **Save** [Button](https://docs.microsoft.com/en-us/dotnet/api/xamarin.forms.button) is pressed the OnSaveButtonClicked event handler is executed, which either saves the content of the Editor to a new file with a randomly generated filename, or to an existing file if a note is being updated. In both cases, the file is stored in the local application data folder for the application. Then the method navigates back to the previous page. When the **Delete** Button is pressed the OnDeleteButtonClicked event handler is executed, which deletes the file, provided that it exists, and navigates back to the previous page. For more information about navigation, see [Navigation](https://docs.microsoft.com/en-us/xamarin/get-started/quickstarts/deepdive#navigation) in the [Xamarin.Forms Quickstart Deep Dive](https://docs.microsoft.com/en-us/xamarin/get-started/quickstarts/deepdive).

Save the changes to **NoteEntryPage.xaml.cs** by pressing **CTRL+S**, and close the file.

1. In **Solution Explorer**, right-click on the **Notes** project and select **Add > New Item..**. In the **Add New Item** dialog, select **Visual C# Items > Xamarin.Forms > Content Page**, name the new file **NotesPage**, and click the **Add** button.

This will add a page named **NotesPage** to the root folder of the project. This page will be the root page of the application.

1. In **NotesPage.xaml**, remove all of the template code and replace it with 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="Notes.NotesPage" Title="Notes"> <ContentPage.ToolbarItems> <ToolbarItem Text="+" Clicked="OnNoteAddedClicked" /> </ContentPage.ToolbarItems> <ListView x:Name="listView" Margin="20" ItemSelected="OnListViewItemSelected"> <ListView.ItemTemplate> <DataTemplate> <TextCell Text="{Binding Text}" Detail="{Binding Date}" /> </DataTemplate> </ListView.ItemTemplate> </ListView></ContentPage> |

This code declaratively defines the user interface for the page, which consists of a [ListView](https://docs.microsoft.com/en-us/dotnet/api/xamarin.forms.listview) and a [ToolbarItem](https://docs.microsoft.com/en-us/dotnet/api/xamarin.forms.toolbaritem). The ListView uses data binding to display any notes that are retrieved by the application, and selecting a note will navigate to the NoteEntryPage where the note can be modified. Alternatively, a new note can be created by pressing the ToolbarItem. For more information about data binding, see [Data binding](https://docs.microsoft.com/en-us/xamarin/get-started/quickstarts/deepdive#data-binding) in the [Xamarin.Forms Quickstart Deep Dive](https://docs.microsoft.com/en-us/xamarin/get-started/quickstarts/deepdive).

This code declaratively defines the user interface for the page, which consists of a [ListView](https://docs.microsoft.com/en-us/dotnet/api/xamarin.forms.listview) and a [ToolbarItem](https://docs.microsoft.com/en-us/dotnet/api/xamarin.forms.toolbaritem). The ListView uses data binding to display any notes that are retrieved by the application, and selecting a note will navigate to the NoteEntryPage where the note can be modified. Alternatively, a new note can be created by pressing the ToolbarItem. For more information about data binding, see [Data binding](https://docs.microsoft.com/en-us/xamarin/get-started/quickstarts/deepdive#data-binding) in the [Xamarin.Forms Quickstart Deep Dive](https://docs.microsoft.com/en-us/xamarin/get-started/quickstarts/deepdive).

Save the changes to **NotesPage.xaml** by pressing **CTRL+S**, and close the file.

1. In **NotesPage.xaml.cs**, remove all of the template code and replace it with the following code:

This code defines the functionality for the NotesPage. When the page appears, the OnAppearing method is executed, which populates the [ListView](https://docs.microsoft.com/en-us/dotnet/api/xamarin.forms.listview) with any notes that have been retrieved from the local application data folder. When the [ToolbarItem](https://docs.microsoft.com/en-us/dotnet/api/xamarin.forms.toolbaritem) is pressed the OnNoteAddedClicked event handler is executed. This method navigates to the NoteEntryPage, setting the [BindingContext](https://docs.microsoft.com/en-us/dotnet/api/xamarin.forms.bindableobject.bindingcontext#Xamarin_Forms_BindableObject_BindingContext) of the NoteEntryPage to a new Note instance. When an item in the ListView is selected the OnListViewItemSelected event handler is executed. This method navigates to the NoteEntryPage, setting the [BindingContext](https://docs.microsoft.com/en-us/dotnet/api/xamarin.forms.bindableobject.bindingcontext#Xamarin_Forms_BindableObject_BindingContext) of the NoteEntryPage to the selected Note instance. For more information about navigation, see [Navigation](https://docs.microsoft.com/en-us/xamarin/get-started/quickstarts/deepdive#navigation) in the [Xamarin.Forms Quickstart Deep Dive](https://docs.microsoft.com/en-us/xamarin/get-started/quickstarts/deepdive).

In **Solution Explorer**, double-click **App.xaml.cs** to open it. Then replace the existing code with the following code:

|  |
| --- |
| using System;using System.IO;using Xamarin.Forms;namespace Notes{ public partial class App: Application { public static string FolderPath {get; private set;} public App() { InitializeComponent(); FolderPath = Path.Combine(Environment.GetFolderPath(Environment.SpecialFolder.LocalApplicationData)); MainPage = new NavigationPage(new NotesPage()); } // ... }} |

* This code adds a namespace declaration for the System.IO namespace, and adds a declaration for a static FolderPath property of type string. The FolderPath property is used to store the path on the device where note data will be stored. In addition, the code initializes the FolderPath property in the App constructor, and initializes the [MainPage](https://docs.microsoft.com/en-us/dotnet/api/xamarin.forms.application.mainpage#Xamarin_Forms_Application_MainPage) property to be a [NavigationPage](https://docs.microsoft.com/en-us/dotnet/api/xamarin.forms.navigationpage) that hosts an instance of NotesPage. For more information about navigation, see [Navigation](https://docs.microsoft.com/en-us/xamarin/get-started/quickstarts/deepdive#navigation) in the [Xamarin.Forms Quickstart Deep Dive](https://docs.microsoft.com/en-us/xamarin/get-started/quickstarts/deepdive).

Save the changes to **App.xaml.cs** by pressing **CTRL+S**, and close the file.

* In **Solution Explorer**, in the **Notes** project, right-click **MainPage.xaml**, and select **Delete**. In the dialog that appears press the **OK** button to remove the file from your hard disk.

This removes a page that's no longer used.

* Build and run the project on each platform. For more information, see [Building the quickstart](https://docs.microsoft.com/en-us/xamarin/get-started/quickstarts/single-page#building-the-quickstart).

On the **NotesPage** press the **+** button to navigate to the **NoteEntryPage** and enter a note. After saving the note the application will navigate back to the **NotesPage**.

Enter a number of notes, of varying length, to observe the application behavior.