Day 9: Android Main Concepts: Intents

Over the next three days, we will explore some of the fundamentals of building an Android Application. They are –

1. Intents
2. Activities
3. Services

Today, let’s explore what Intents in Android are –

Intents in Android are just data structures that are message objects. Intents can request an operation to be performed by some other component in Android. Intents are generally used to start **Activities** and **Services** in Android. We will explore in coming blog posts on how to use Intents to start an Activity and Services.

All Intents are classified into two Types –

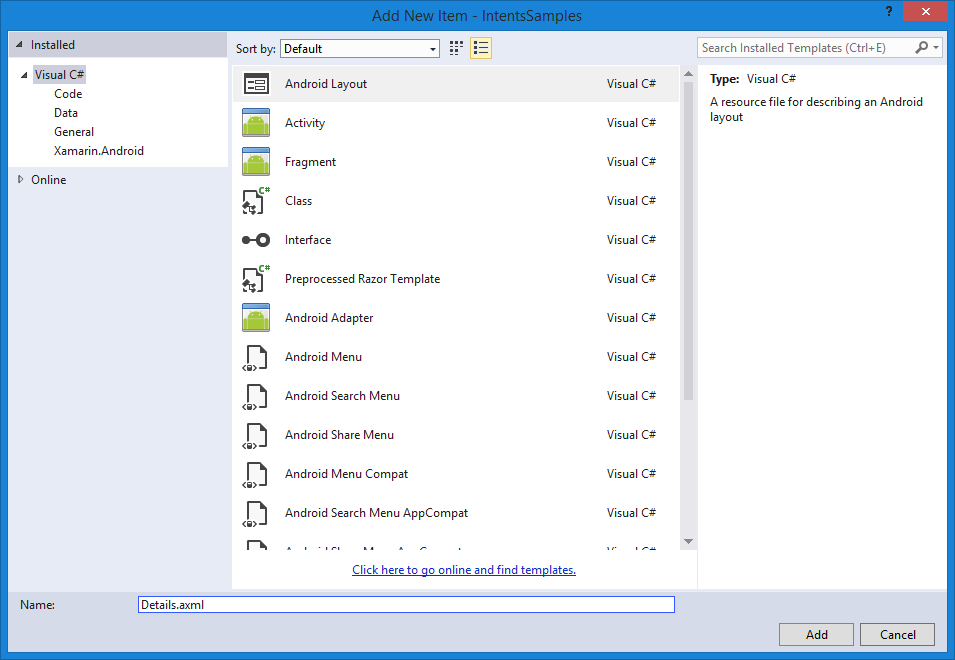
1. Explicit Intents
2. Implicit Intents

## Explicit Intents

Explicit Intents are where you specify Component that needs to be launched by its class name. The common use case for this is, when, you want to start an activity in your own Android App.

Let’s look at an example of Explicit Intent.

In our Simple Hello World Android Application, let’s add another Layout first –



Let’s also go ahead and add an Activity (You will learn what activities are in the next post, but for now please note that each screen in Android corresponds to an Activity. This Activity will work with the Details Layout that we just added –

|  |
| --- |
| using Android.App;  using Android.OS;  namespace IntentsSamples  {  [Activity(Label = "DetailsActivity")]  public class DetailsActivity : Activity  {  protected override void OnCreate(Bundle bundle)  {  base.OnCreate(bundle);  SetContentView(Resource.Layout.Details);  }  }  } |

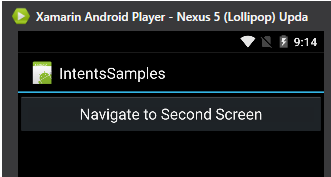
Gist file link: <https://gist.github.com/vkoppaka/bd9cfe6a9f09d77ec67b>

Now, let’s go back to our main Activity and write some code to make so that the button click on the activity takes us to newly created Details Activity.

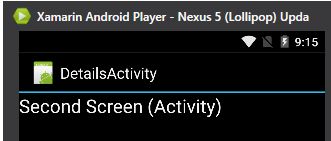
|  |
| --- |
| using System;  using Android.App;  using Android.Content;  using Android.OS;  using Android.Widget;  namespace IntentsSamples  {  [Activity(Label = "IntentsSamples", MainLauncher = true, Icon = "@drawable/icon")]  public class MainActivity : Activity  {  protected override void OnCreate(Bundle bundle)  {  base.OnCreate(bundle);  // Set our view from the "main" layout resource  SetContentView(Resource.Layout.Main);  // Get our button from the layout resource,  // and attach an event to it  Button button = FindViewById<Button>(Resource.Id.MyButton);  button.Click += button\_Click;  }  private void button\_Click(object sender, EventArgs e)  {  var detailsIntent = new Intent(this, typeof (DetailsActivity));  StartActivity(detailsIntent);  }  }  } |

Gist file link: <https://gist.github.com/vkoppaka/2f179167aea3693761b0>

Notice in the button\_Click method that we are constructing an Intent and then passing the details of the Activity we are hoping to move to (fully qualified name). And once an Intent is constructed, we just call “StartActivity” method with the intent as a parameter.



Main screen above, and once Navigate to Second Screen button is pressed then we move to Details Activity (Second Screen)



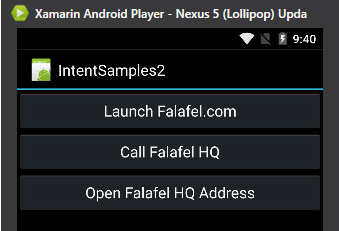
## Implicit Intents

Implicit Intents don’t specify Component but rather define the action that needs to be performed. This actions are then handled by other apps in Android (be it be built-in or third party apps).

Some of the common examples for Implicit Intents are –

1. Opening a Web Page
2. Launching Maps
3. Launching Phone Dialer

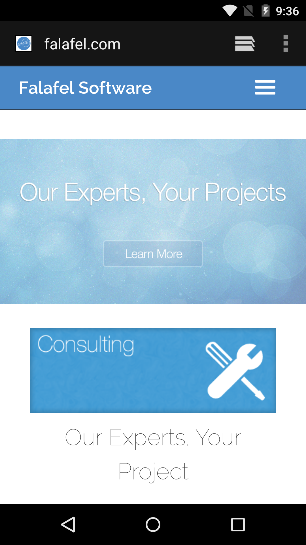
Let’s build a quick little sample that demonstrates all these



### Opening a Web Page

To open a Web Page with Intents, you would use an Intent.ActionView action and pass URL message object. Let’s see it in action –

|  |
| --- |
| void launchFalafelComButton\_Click(object sender, EventArgs e)  {  var uri = Uri.Parse("http://falafel.com");  var intent = new Intent(Intent.ActionView, uri);  StartActivity(intent);  } |



### Launching Maps

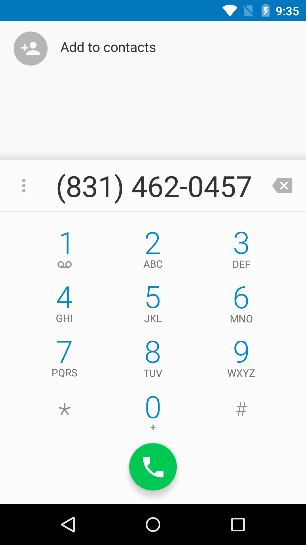
To launch maps with Intents, you would use Intent.ActionView action and pass a Geo Location message object. Let’s see it in action –

|  |
| --- |
| private void openFalafelHqAddressButton\_Click(object sender, EventArgs e)  {  var uri = Uri.Parse("geo:37.0634800,-122.0013280");  var intent = new Intent(Intent.ActionView, uri);  StartActivity(intent);  } |

### Launching Phone Dialer

To launch phone dialer with Intents, you would use Intent.ActionDial action and pass a telephone number message object. Let’s see it in action –

|  |
| --- |
| private void callFalafelHqButton\_Click(object sender, EventArgs e)  {  var uri = Uri.Parse("tel:8314620457");  var intent = new Intent(Intent.ActionDial, uri);  StartActivity(intent);  } |



Gist file link: <https://gist.github.com/vkoppaka/ecebcace923869a57f30>

That’s it for today, tomorrow, let’s explore what Activities in Android are in more detail.

Venkata