**Learning Outcomes**

1. Understanding ActionBarSherlock

**ActionBars**

The action bar is a window feature that identifies the user location, and provides user actions and navigation modes. Using the action bar offers your users a familiar interface across applications that the system gracefully adapts for different screen configurations.

The [ActionBar](http://developer.android.com/reference/android/app/ActionBar.html) APIs were first added in Android 3.0 (API level 11) but they are also available in the [Support Library](http://developer.android.com/tools/support-library/index.html) for compatibility with Android 2.1 (API level 7) and above.

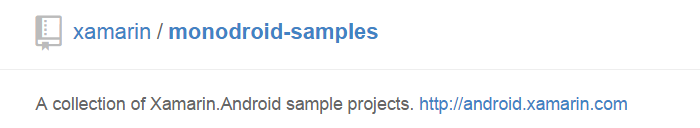


**Sherlock Action Bar**

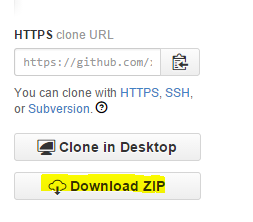
ActionBarSherlock is a Java Android library that brings action bar feature, which became available in Android 4.0 API, to Android 2.x too.

Download the sample code from github for the Sherlock action bar and go through its features

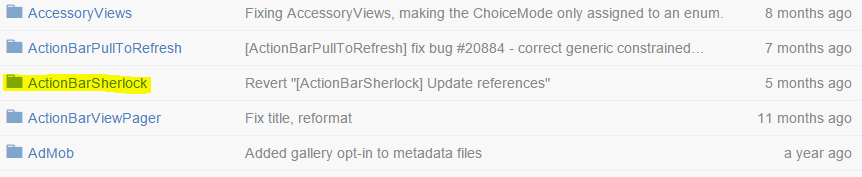
On Github search for

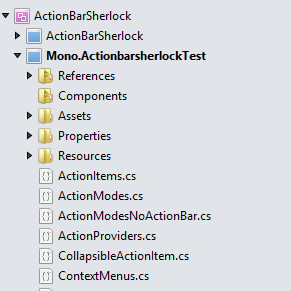


Download the entire ZIP file.



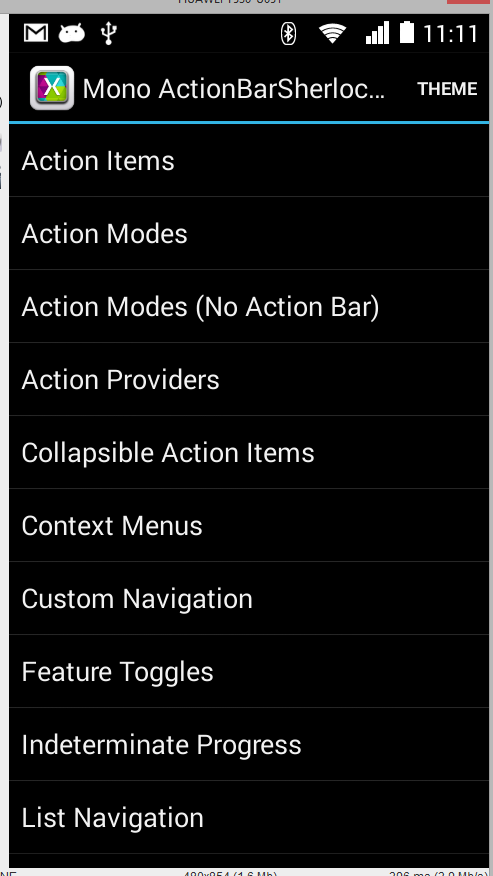
Extract Action Bar Sherlock and run the solution.





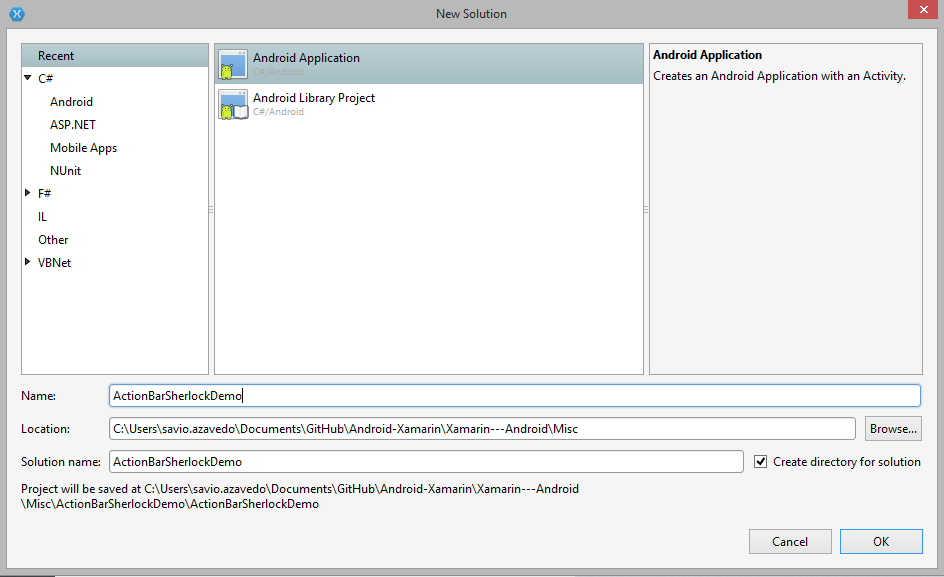
At the top you have the Actionbar Sherlock Library and then the Mono.ActionbarsherlockTest project.

Build and Run the Action bar Sherlock project. (Play around with the app and understand its features)

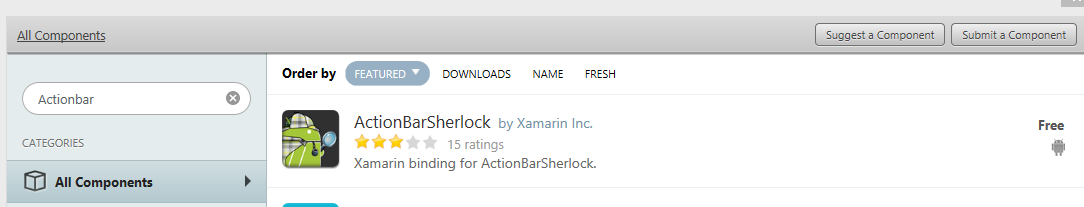


We will try to add some of the features to our project.

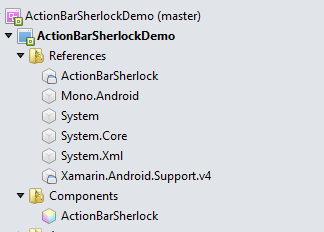
Create a New Solution name it ActionBarSherlockDemo



Search for Actionbar Sherlock in components and add it to your project.



When you add the component it automatically adds the reference to the library.

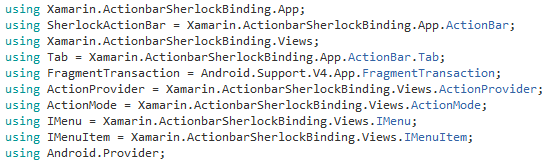


Before doing anything just build the complete project.

1. **Adding Action Items**

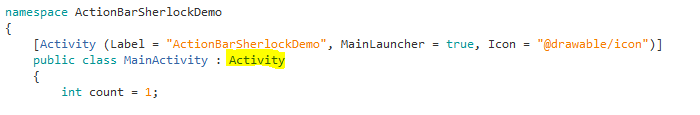
We will add some action items on the ActionBar.

In MainActivity.cs import a bunch of libraries.

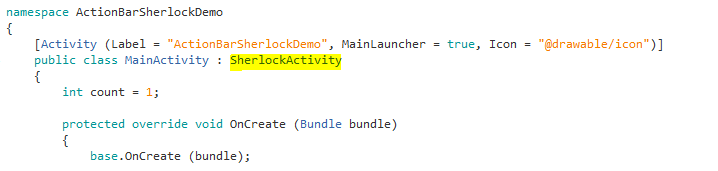


Copy it down from the ActionBarSherlockTest project.

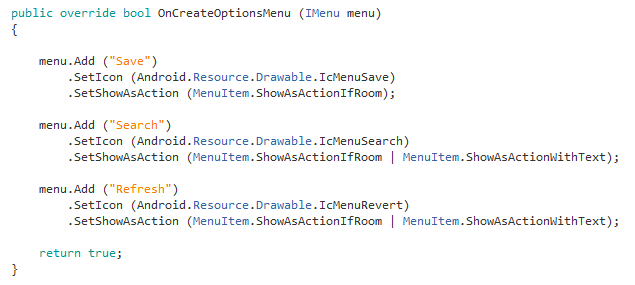
Next important change is to inherit the Activity as SherlockActivity.



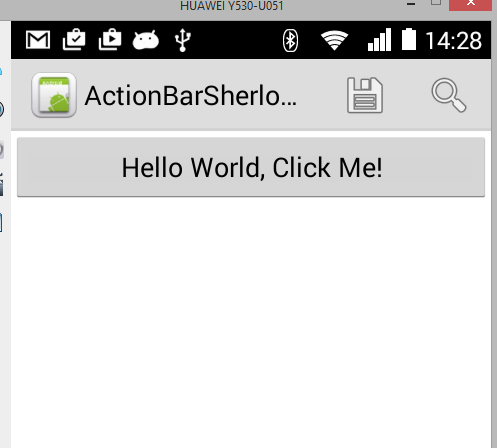
**To**



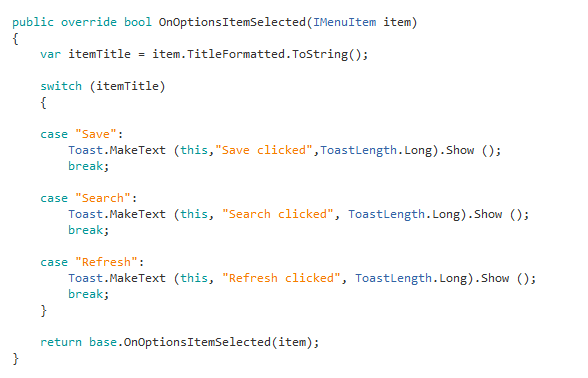
Now to add ActionItems on the ActionBar, write the following code on the OnCreateOptionsMenu

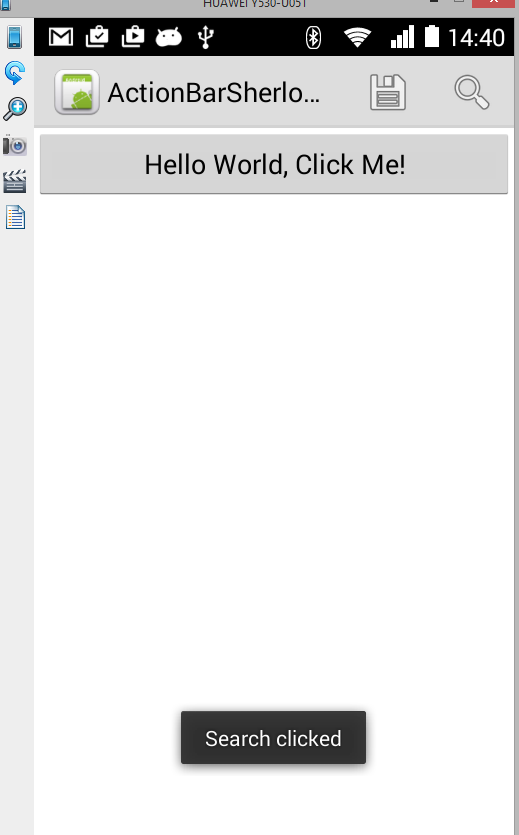


Run the program to get the following output.



Handling the click event of the Action Items.

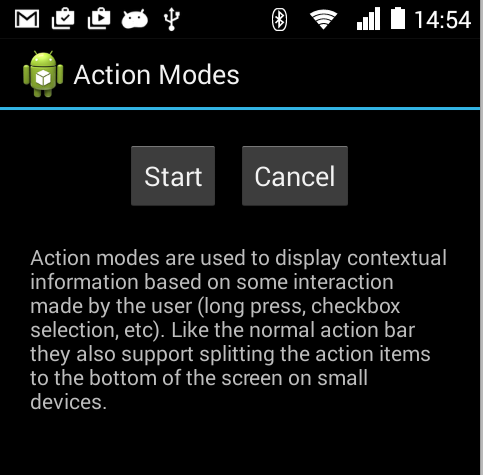
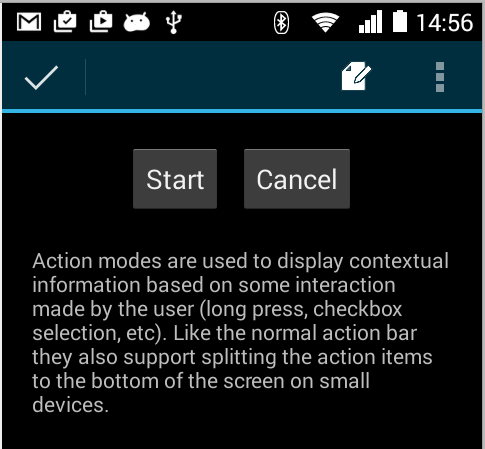




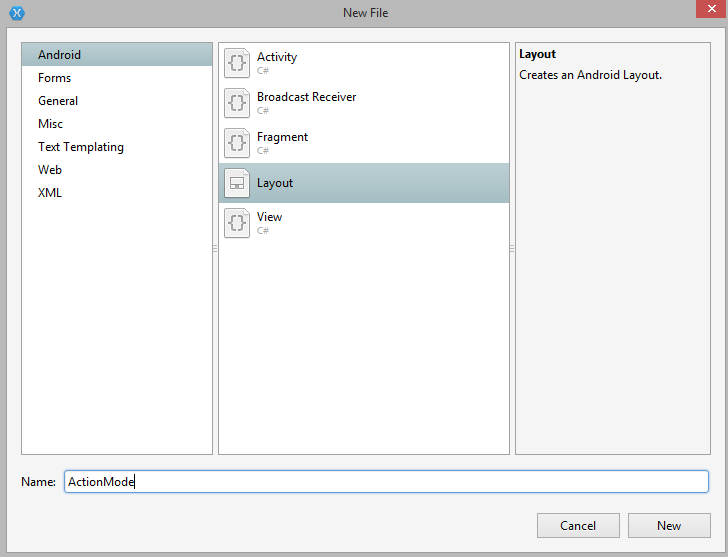
**Action Modes**

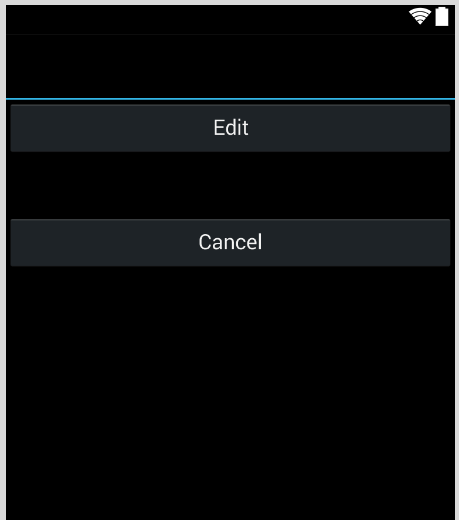
Represents a contextual mode of the user interface. Action modes can be used to provide alternative interaction modes and replace parts of the normal UI until finished. Examples of good action modes include text selection and contextual actions.

**Normal Mode Action Mode**

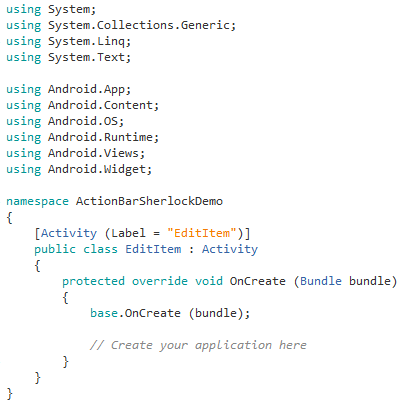
 

Let us start by creating a new layout and put two buttons on the same





Create a new Activity **EditItem.cs**



1. Change your Activity to SherlockActivity
2. Copy the library imports given previously.

Next we need to have a Nested Class as shown which inherits from

Java.Lang.Object, ActionMode.ICallback

