Using Location Service APIs in WP7

This article explains how to use Location Service APIs (GeoCoordinateWatcher) in WP7.

Introduction

To use the WP7 Location Service APIs you need to add a reference to `System.Device.dll`.

The following code snippet shows how to initialize the location service, handle changes in the service's status and obtain location data on WP7.

```
public partial class MainPage : PhoneApplicationPage
{
   GeoCoordinateWatcher watcher;
   // Click the event handler for the “Start Location” button.
   private void startLocationButton_Click(object sender, RoutedEventArgs e)
   {
      // The watcher variable was previously declared as type GeoCoordinateWatcher.
      if (watcher == null)
      {
         watcher = new GeoCoordinateWatcher(GeoPositionAccuracy.High);  // using high
accuracy
         watcher.MovementThreshold = 20; // use MovementThreshold to ignore noise in the
signal
         watcher.StatusChanged += new
EventHandler<GeoPositionStatusChangedEventArgs>(watcher_StatusChanged);
         watcher.PositionChanged += new
EventHandler<GeoPositionChangedEventArgs<GeoCoordinate>>(watcher_PositionChanged);
      }
      watcher.Start();
} // End of the Start button Click handler.


// Event handler for the GeoCoordinateWatcher.StatusChanged event.
// Implementation of the StatusChanged event handler.
// This event is raised whenever the status of the Location Service changes.
// The GeoPositionStatus enumeration that is passed in the
GeoPositionStatusChangedEventArgs object,
// tells you the current status of the service.

void watcher_StatusChanged(object sender, GeoPositionStatusChangedEventArgs e)
{
    switch (e.Status)
    {
        case GeoPositionStatus.Disabled:
            // The Location Service is disabled or unsupported.
            // Check to see whether the user has disabled the Location Service.
            if (watcher.Permission == GeoPositionPermission.Denied)
            {
                // The user has disabled the Location Service on their device.
                statusTextBlock.Text = "you have this application access to location.";
            }
            else
            {
                statusTextBlock.Text = "location is not functioning on this device";
            }
```


(C) Copyright Nokia 2014. All rights reserved.

```
            break;

        case GeoPositionStatus.Initializing:
            // The Location Service is initializing.
            // Disable the Start Location button.
            startLocationButton.IsEnabled = false;
            break;
        case GeoPositionStatus.NoData:
            // The Location Service is working, but it cannot get location data.
            // Alert the user and enable the Stop Location button.
            statusTextBlock.Text = "location data is not available.";
            stopLocationButton.IsEnabled = true;
            break;

        case GeoPositionStatus.Ready:
            // The Location Service is working and is receiving location data.
            // Show the current position and enable the Stop Location button.
            statusTextBlock.Text = "location data is available.";
            stopLocationButton.IsEnabled = true;
            break;
    }
}

//When the Location Service is ready and receiving data, it will begin to raise the
// PositionChanged event and call your application’s handler if you have implemented
one.
// In the event handler, access the Position member of the
GeoPositionChangedEventArgs(Of T) object.
// The Position field is a GeoPosition object, which consists of a Timestamp and a
GeoCoordinate
// object that contains the location information for the reading. This example accesses
the latitude and longitude values.

void watcher_PositionChanged(object sender, GeoPositionChangedEventArgs<GeoCoordinate>
e)
{
  latitudeTextBlock.Text = e.Position.Location.Latitude.ToString("0.000");
  longitudeTextBlock.Text = e.Position.Location.Longitude.ToString("0.000");
}

// Click the event handler for the “Start Location” button.
private void stopLocationButton_Click(object sender, RoutedEventArgs e)
{
  watcher.Stop();
}
```


(C) Copyright Nokia 2014. All rights reserved.

(C) Copyright Nokia 2014. All rights reserved.