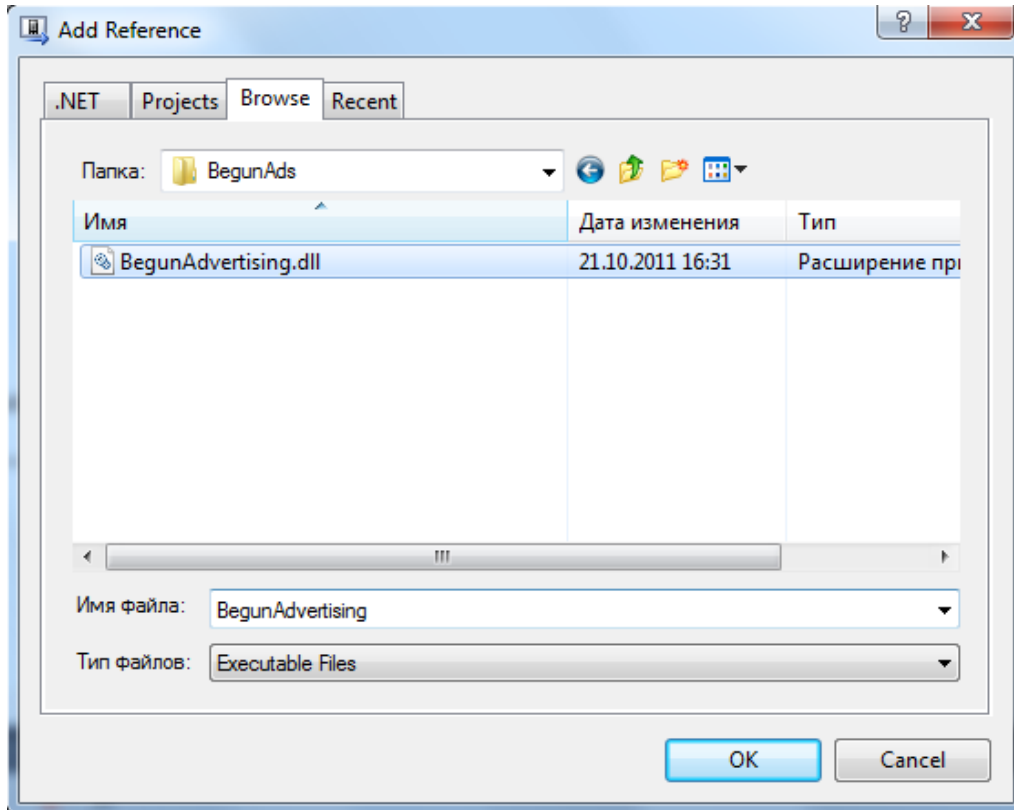


## Бегун. Установка рекламной библиотеки на Windows Phone 7

Перед установкой необходимо убедиться, что Бегун предоставил Вам **PadId** и **BlockId**. Эти параметры необходимы для корректной инициализации библиотеки.

- 1) Через меню **Project -> Add Reference** необходимо добавить библиотеку **BegunAdvertising.dll** в проект.



- 2) Необходимо объявить пространство имен библиотеки в приложении. Для этого нужно внести изменения в **MainPage.xaml**

```
<phone:PhoneApplicationPage
    xmlns:AdvControl="clr-namespace:Begun.Ads.WindowsPhone7;assembly=BegunAdvertising">
</phone:PhoneApplicationPage>
```

Так же необходимо добавить описание самой библиотеки в **MainPage.xaml**

```
<AdvControl:BegunAdvertising x:Name="Advertising" BlockId="{BLOCK_ID}" PadId="{PAD_ID}"
    Width="320" Height="80">
</AdvControl:BegunAdvertising>
```

Альтернативный способ **подключения библиотеки - программным способом:**

```
using Begun.Ads.WindowsPhone7;

BegunAdvertising adv = new BegunAdvertising();
adv.Width = 480;
adv.Height = 80;
adv.PadId = "{PAD_ID}";
adv.BlockId = "{BLOCK_ID}";
ContentPanel.Children.Add(adv);
```

- 3) Библиотека позволяет получать уведомления о своем состоянии. Таким образом разработчик всегда будет в курсе, когда произошел переход по отображаемому рекламному объявлению. Для этого необходимо добавить обработчик событий библиотеки **AdEvent**:

```

Advertising.AdvEvent += new Begun.Ads.WindowsPhone7.BegunAdvEventHandler(Advertising_AdvEvent);

/**
 * Example handler
 */
void Advertising_AdvEvent(object source, Begun.Ads.WindowsPhone7.BegunAdvEventArgs args)
{
    Debug.WriteLine(args.EventType + " : " + args.EventMessage);
    switch (args.EventType)
    {
        case "AdLoaded":
            _isAdLoaded = true;
            break;
    }
}

```

На текущий момент доступны следующие типы уведомлений `args.EventType`:

- AdInit – инициализация библиотеки
- AdLoaded – инициализация прошла успешно
- AdStarted – библиотека начала работу
- AdStopped – библиотека закончила работу
- AdClickThru – произошло событие перехода по рекламному объявлению
- AdError – произошло исключение при работе библиотеки

В случае необходимости использования в приложении более одной компоненты библиотеки, необходимо принудительно осуществить запуск и остановку показа рекламных блоков библиотеки. Для таких целей необходимо использовать соответствующие методы:

```

Advertising.stopAd();

if (_isAdLoaded)
{
    Advertising.startAd();
}

```

Результат интеграции **рекламного блока** от Бегуна:



**Мобильный сайт знакомств!**  
Мобильный сайт знакомств. Заходите с мобильного.



4) В случае, если ваше приложение использует геолокационные возможности устройства, библиотека позволит осуществить более точный таргетинг в зависимости от текущего местоположения пользователя. Для передачи информации о местоположении пользователя Вами должен быть реализован класс, в основе которого лежит компонента **GeoCoordinateWatcher**. Примером может быть код, приведенный ниже:

```
using System;
using System.Device.Location;

namespace Begun.Ads.WindowsPhone7
{
    public delegate void BegunGpsEventHandler();
    public class GpsLocationProvider
    {
        public static event BegunGpsEventHandler GpsPositionChanged;
        static GpsLocationProvider()
        {
            try
            {
                CurrentLocation = null;
                GeoCoordinateWatcher Provider = new GeoCoordinateWatcher();
                Provider.PositionChanged += new EventHandler
                    <GeoPositionChangedEventArgs<GeoCoordinate>>
                    (GpsLocationProvider_PositionChanged);
                Provider.Start(true);
            }
            catch (Exception exception)
            {
                System.Diagnostics.Debug.WriteLine(exception);
            }
        }
        private static void GpsLocationProvider_PositionChanged(object sender,
            GeoPositionChangedEventArgs<GeoCoordinate> e)
        {
            if (e.Position.Location.IsUnknown)
                CurrentLocation = null;
            else
            {
                CurrentLocation = new GpsLocation
                {
                    Latitude = e.Position.Location.Latitude,
                    Longitude = e.Position.Location.Longitude
                };
                if (GpsPositionChanged != null)
                {
                    GpsPositionChanged();
                }
            }
        }
        public static GpsLocation? CurrentLocation
        {
            get;
            private set;
        }
    }
}
```

Вам необходимо передать библиотеке объект **GpsLocation**.

```
GpsLocationProvider.GpsPositionChanged += new
Begun.Ads.WindowsPhone7.BegunGpsEventHandler(GpsLocationProvider_GpsPositionChanged);

void GpsLocationProvider_GpsPositionChanged()
{
    Advertising.GpsLocation = Begun.Ads.WindowsPhone7.GpsLocationProvider.CurrentLocation;
}
```