Ad in Xamarin.Forms App using AdMob

1. Create Ad ID(s) on AdMob Website for each Platform

2. iOS:
   a. Add AdMob Component from Xamarin Component Store

3. Droid:
   a. Add Google Play Services Component from Xamarin Component Store

4. WinPhone:
   a. Download and Extract AdMob SDK from AdMob Website
      (https://developers.google.com/admob/wp/quick-start)
   b. Right Click on WinPhone Project Add > References > Navigate to dll file (unblock the file
      before referencing http://blogs.msdn.com/b/delay/p/unblockingdownloadedfile.aspx)

5. Create an empty class inheriting 'View' in Portable Project
```
using System;
using System.Collections.Generic;
using System.Linq;
using System.Text;
using System.Threading.Tasks;
using Xamarin.Forms;
namespace MyApp
{
    public class AdControl : View
    {

    }
}
```

6. Create a class in each Platform inheriting 'ViewRenderer'

   iOS:
```
using System;
using System.Collections.Generic;
using System.Linq;
using System.Text;
using Foundation;
using UIKit;
using Xamarin.Forms.Platform.iOS;
using Xamarin.Forms;
using MyApp;
using MyApp.iOS;
using GoogleAdMobAds;

[assembly: ExportRenderer(typeof(AdControl), typeof(AdControliOS))]
namespace MyApp.iOS
```

```
    {
        class AdControliOS : ViewRenderer
        {
            GADBannerView adView;
            bool viewOnScreen = false;

            protected override void
    OnElementChanged(ElementChangedEventArgs<Xamarin.Forms.View> e)
            {
                base.OnElementChanged(e);

                adView = new GADBannerView(size: GADAdSizeCons.Banner)//,
    origin: new PointF(0, 0))
                {
                    AdUnitID = "YOUR-KEY",
                    RootViewController =
    UIApplication.SharedApplication.Windows[0].RootViewController,
                };

                adView.AdReceived += (sender, args) =>
                {
                    if (!viewOnScreen) this.AddSubview(adView);
                    viewOnScreen = true;
                };

                adView.LoadRequest(GADRequest.Request);
                base.SetNativeControl(adView);
            }
        }
    }


Droid:
using System;
using System.Collections.Generic;
using System.Linq;
using System.Text;
using Android.Gms.Ads;
using MyApp;
using MyApp.Droid;
using Xamarin.Forms;
using Xamarin.Forms.Platform.Android;

[assembly: ExportRenderer(typeof(AdControl), typeof(AdControlAndroid))]
namespace MyApp.Droid
{
    class AdControlAndroid : ViewRenderer
    {
        protected override void
    OnElementChanged(ElementChangedEventArgs<View> e)
        {
            base.OnElementChanged(e);

            if (e.OldElement == null)
            {
                AdView ad = new AdView(this.Context);
                ad.AdSize = AdSize.Banner;
```

Pranav Sharma

```
                    ad.AdUnitId = "YOUR-KEY";
                    var requestbuilder = new AdRequest.Builder();
                    ad.LoadAd(requestbuilder.Build());
                    this.SetNativeControl(ad);
                }
            }
        }
    }
```

WinPhone:
```
using System;
using System.Collections.Generic;
using System.Linq;
using System.Text;
using System.Threading.Tasks;
using MyApp;
using MyApp.WinPhone;
using GoogleAds;
using Xamarin.Forms.Platform.WinPhone;
using Xamarin.Forms;

[assembly: ExportRenderer(typeof(AdControl), typeof(AdControlWinPhone))]
namespace MyApp.WinPhone
{
    class AdControlWinPhone : ViewRenderer
    {
        protected override void
OnElementChanged(ElementChangedEventArgs<View> e)
        {
            base.OnElementChanged(e);
            if (e.OldElement == null)
            {
                AdView bannerAd = new AdView
                {
                    Format = AdFormats.Banner,
                    AdUnitID = "YOUR-KEY",
                };
                AdRequest adRequest = new AdRequest();
                bannerAd.LoadAd(adRequest);
                Children.Add(bannerAd);
            }
        }
    }
}
```

7. Using live ads during development is against AdMob policy; if you test on live ads, your AdMob account may be suspended.
    a. How do I get test ads on my physical device?
       Use ForceTesting to enable test ads.

8. Edit Android Manifest:
    a. ACCESS_NETWORK_STATE
    b. INTERNET

Pranav Sharma

c. Add this between opening and closing application tags

```
<application ………>
    …
    …
    <meta-data android:name="com.google.android.gms.version"
    android:value="@integer/google_play_services_version" />
    <activity android:name="com.google.android.gms.ads.AdActivity"
    android:configChanges="keyboard|keyboardHidden|orientation|screenLayout|uiMode|screenSize|smallestScreenSize" />
    …
    …
    …
</application>
```

9. Edit WinPhone Manifest:
   a. ID_CAP_NETWORKING
   b. ID_CAP_WEBBROWSERCOMPONENT
   c. ID_CAP_MEDIALIB_PLAYBACK
   d. ID_CAP_MEDIALIB_AUDIO

Pranav Sharma