Building an App for Android and iOS with Xamarin & C#

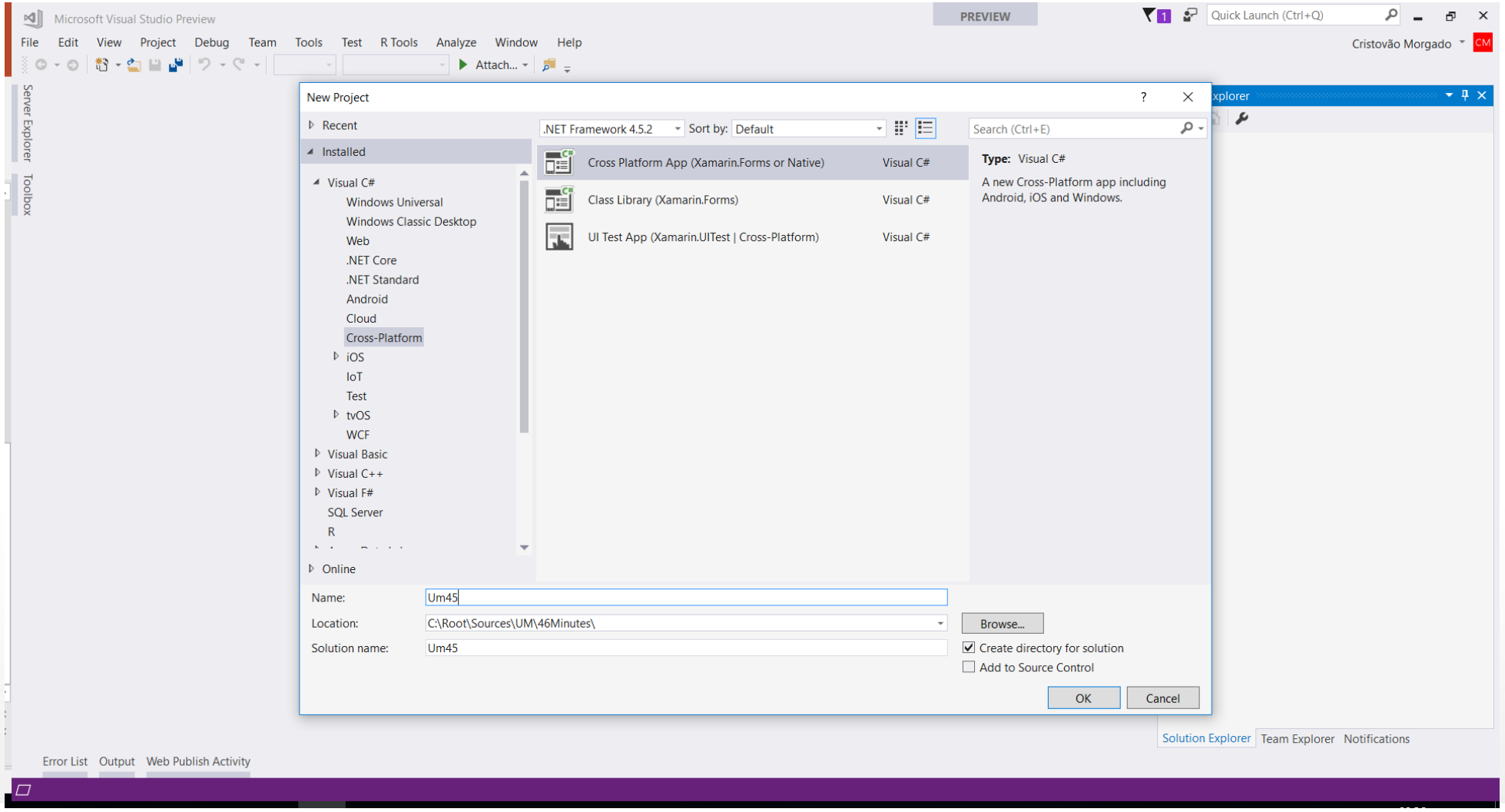
Under one hour!!

Cristóvão Morgado

Pre requisites

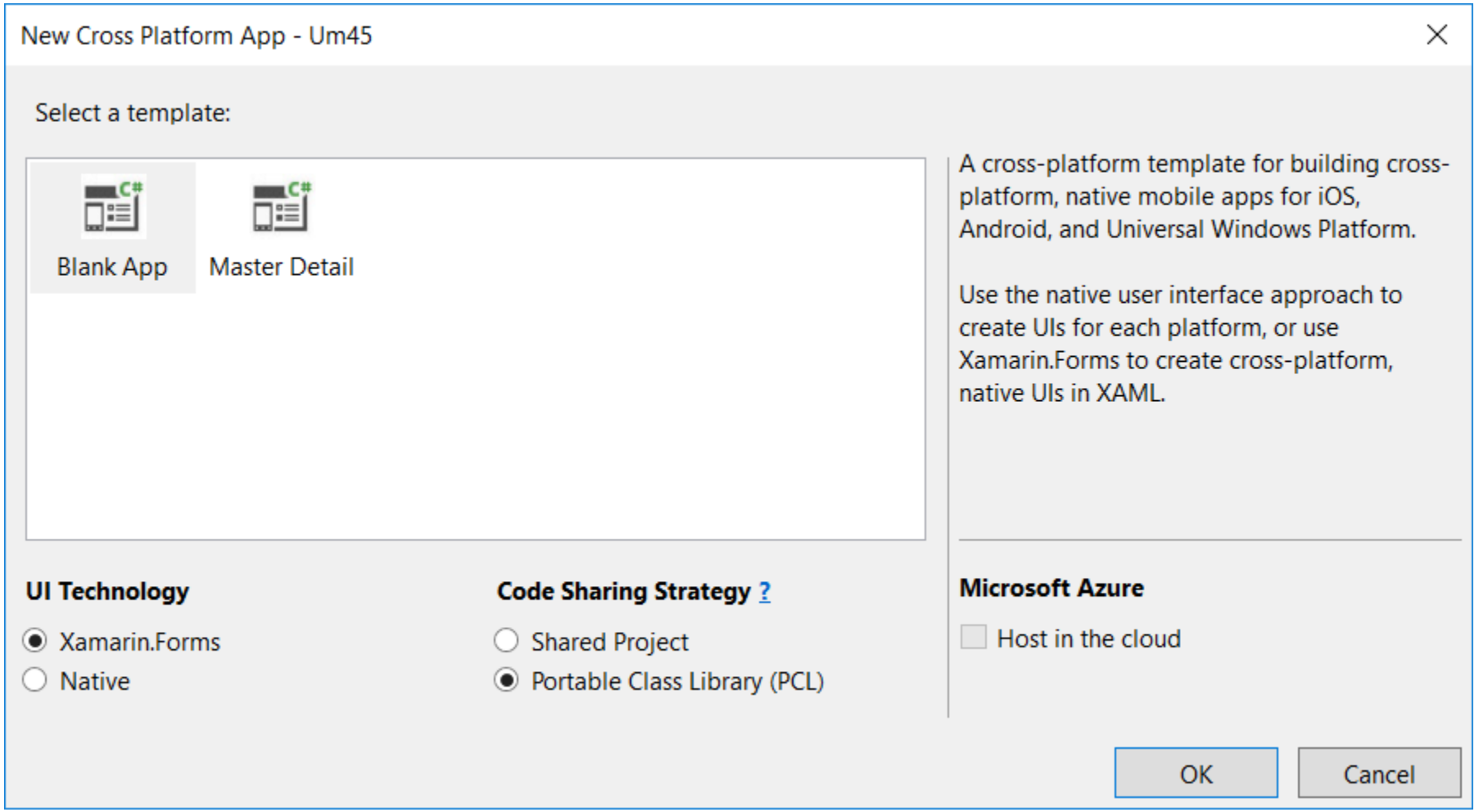
1. C# Know-how
2. Windows 10 Creators Update
3. Min 8Gb RAM
4. Visual Studio 2017 Preview (15.3 (>26510.0-Preview))
    1. Xamarin

Choosing the right project template

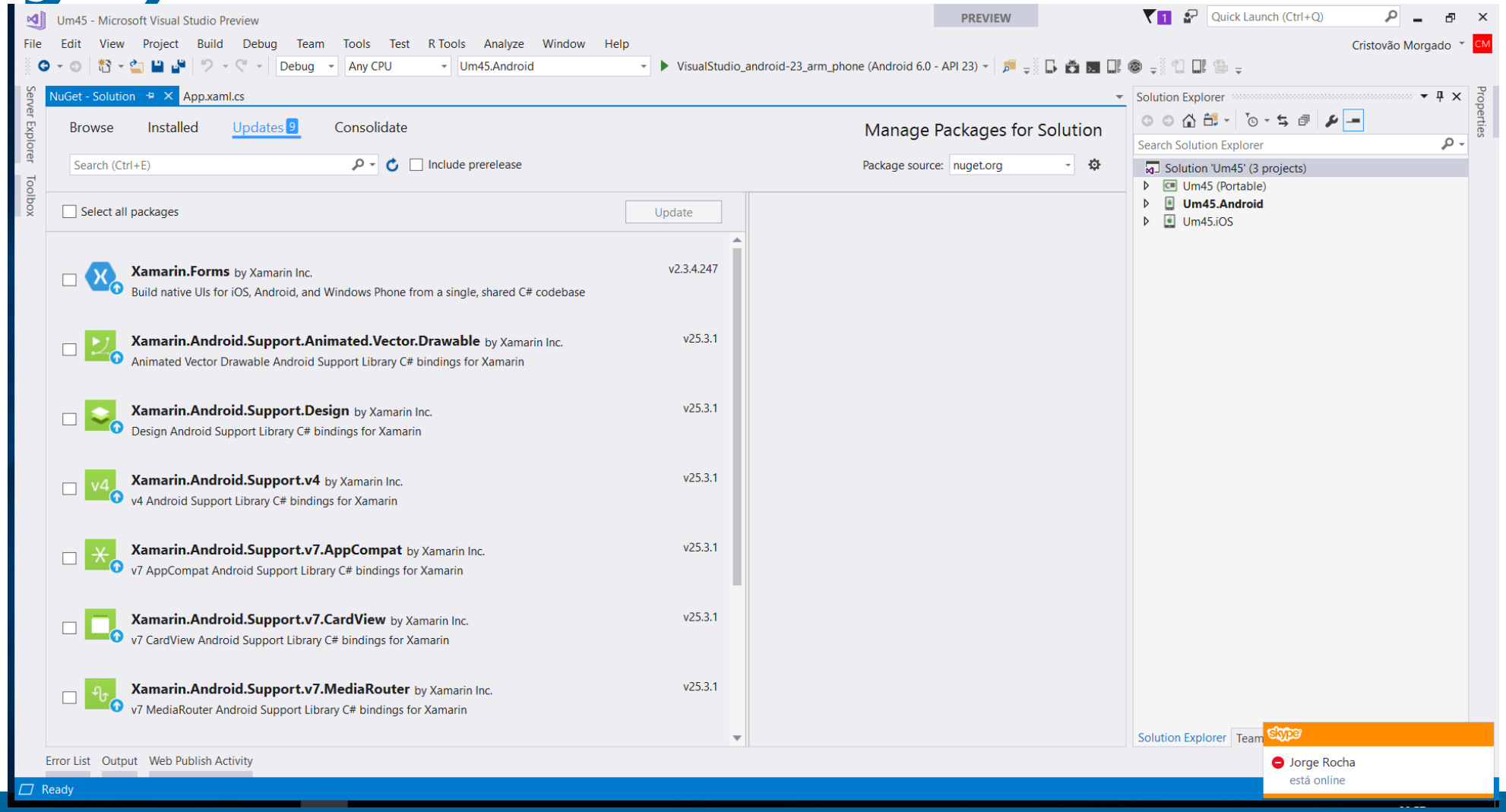


IT sector
Sistemas de Informação, SA

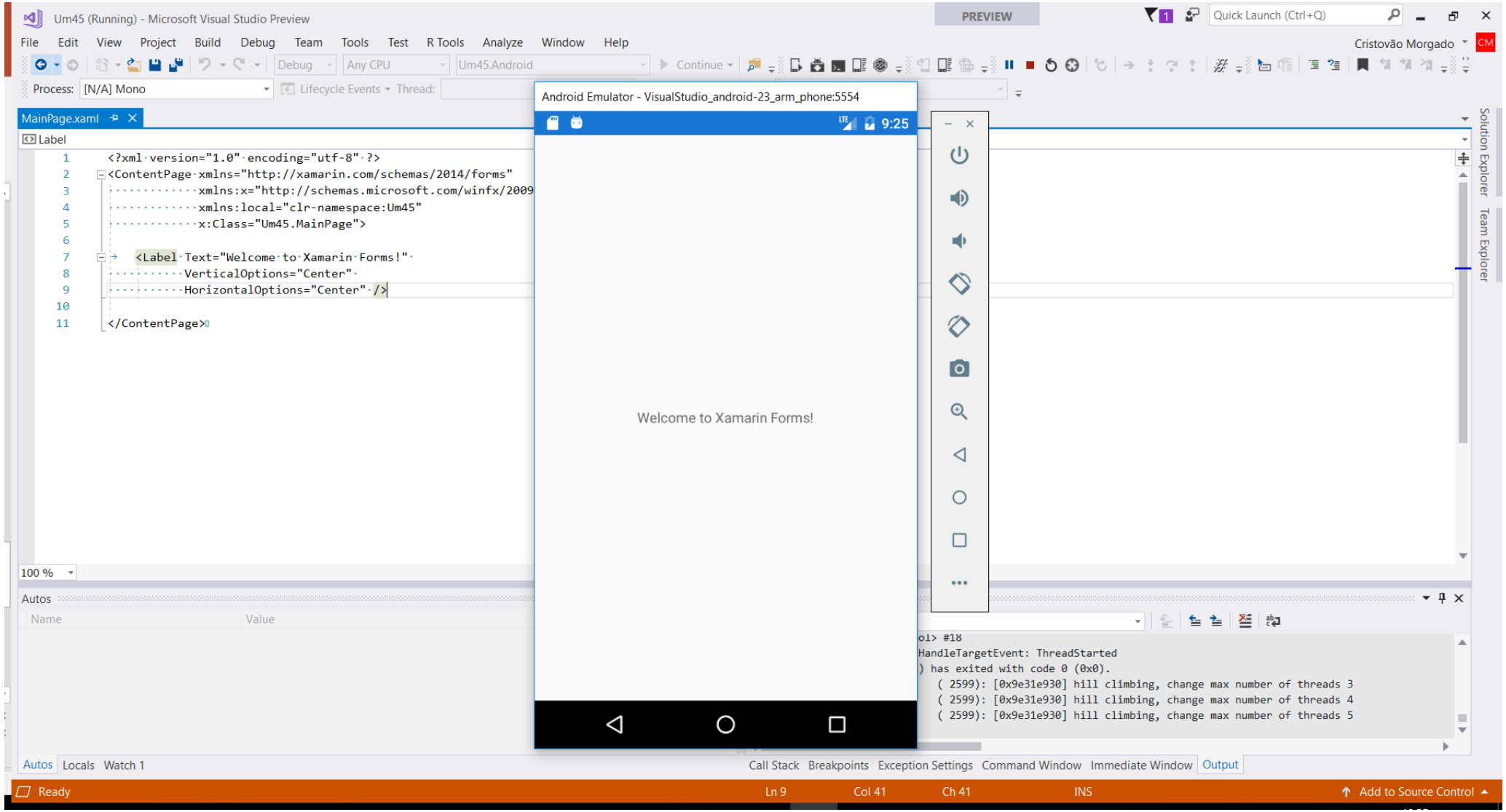
The barebone template



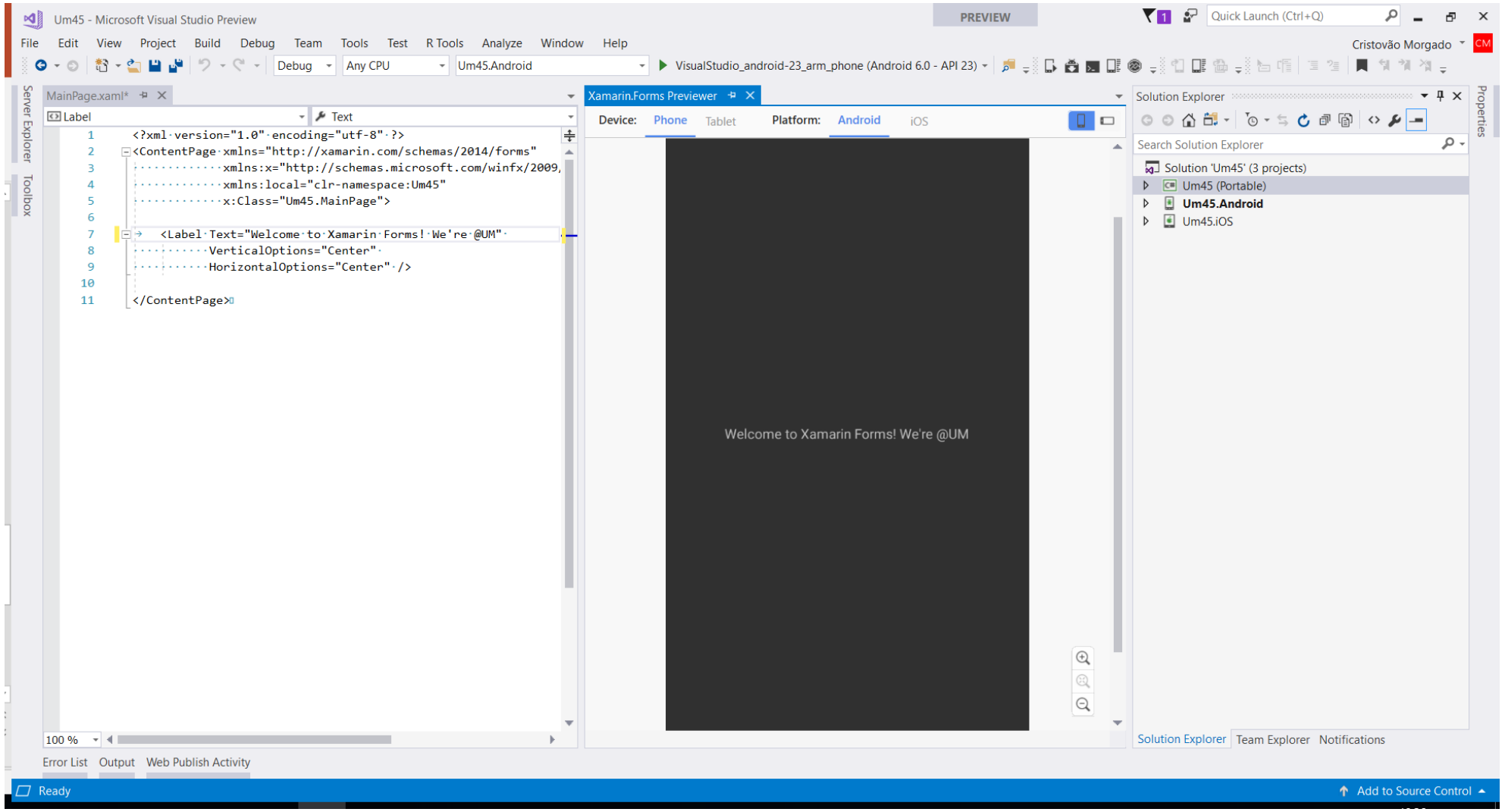
Before starting just update all dependencies (packages)



Let's run the barebone app in the Android Emulator

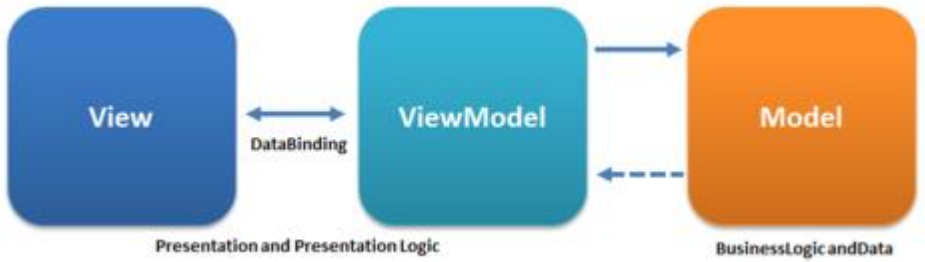


Using the Xamarin Forms Previewer

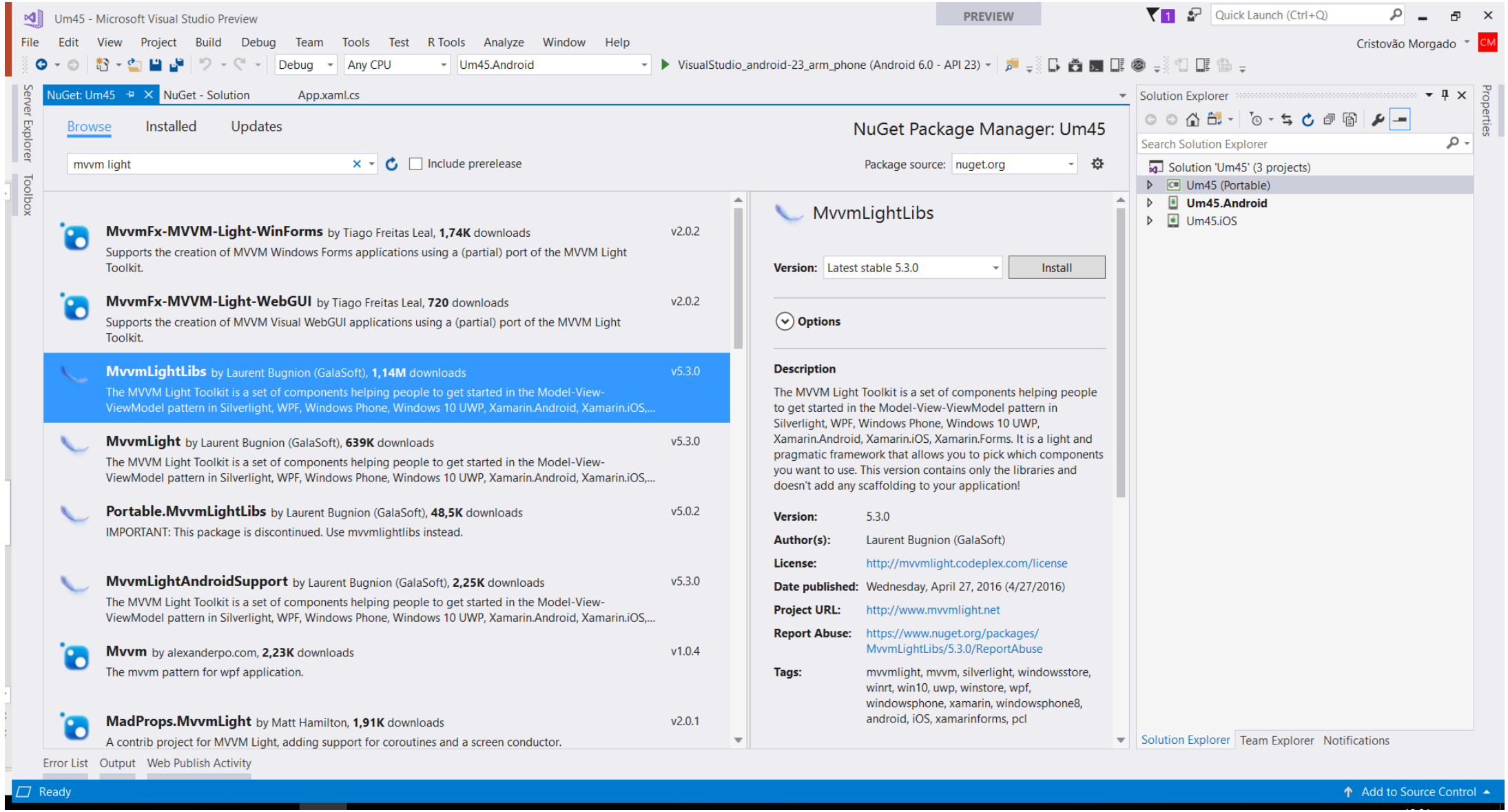


MVVM definition (source Wikipedia)

Model–view–viewmodel (MVVM) is a software architectural pattern.
MVVM facilitates a separation of development of the graphical user interface – be it via a markup language or GUI code – from development of the business logic or back-end logic (the *data model*).
The *view model* of MVVM is a value converter;[1] meaning the view model is responsible for exposing (converting) the data objects from the model in such a way that objects are easily managed and presented.
In this respect, the view model is more model than view, and handles most if not all of the view's display logic.[1]
The view model may implement a mediator pattern,
organizing access to the back-end logic around the set of use cases supported by the view.



The best library for MVVM ?! MVVM Light



INotifyPropertyChanged

```
using System;
using System.ComponentModel;
using System.Runtime.CompilerServices;

namespace Um45.Models
{
    2 references
    public class ModelBase : INotifyPropertyChanged
    {
        public event PropertyChangedEventHandler PropertyChanged;
        9 references
        public void NotifyPropertyChanged([CallerMemberName]String propertyName = null)
        {
            var handler = PropertyChanged;
            if (null == handler) return;
            try
            {
                handler(this, new PropertyChangedEventArgs(propertyName));
            }
            catch
            {
                //ignored
            }
        }
    }
}
```

Model

```
namespace Um45.Models
{
    13 references
    public class GitHubGuy:ModelBase
    {
        private string _name;
            7 references
        public string Name

        {
            get => _name;
            set
            {
                if (_name == value) return;
                _name = value;
                NotifyPropertyChanged();
            }
        }

        private string _avatar;
            7 references
        public string Avatar

        {
            get => _avatar;
            set
            {
                if (_avatar == value) return;
                _avatar = value;
                NotifyPropertyChanged();
            }
        }
    }
}
```

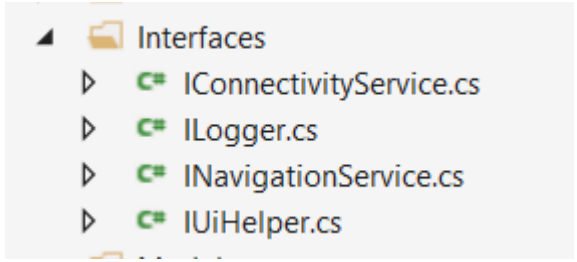
ViewModels

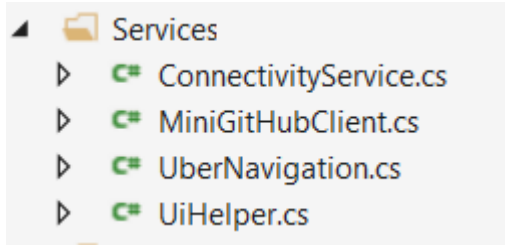
```
4 references
public class HomeViewModel : BaseViewModel
{
    0 references
    public HomeViewModel(IConnectivityService connectivityService
        , IUiHelper uiHelper
        , INavigationService navigationService
        , ILogger logger) : base(connectivityService, uiHelper, navigationService, logger)
    {
        ...
    }

    private ObservableCollection<GitHubGuy> _items = new ObservableCollection<GitHubGuy>();
    2 references
    public ObservableCollection<GitHubGuy> Items

    {
        get => _items;
        set
        {
            if (_items == value) return;
            _items = value;
            NotifyPropertyChanged();
        }
    }
```

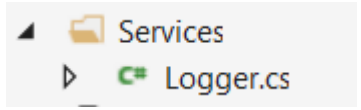
Binding

```
    <Grid  Grid.Row="1" >
        <Grid.ColumnDefinitions>
            <ColumnDefinition Width="*"></ColumnDefinition>
            <ColumnDefinition Width="Auto"></ColumnDefinition>
        </Grid.ColumnDefinitions>
        <Entry  Margin="10" HorizontalOptions="FillAndExpand" BackgroundColor="White"
                Text="{Binding SearchQuery, Mode=TwoWay}"></Entry>
        <Button Grid.Column="1" Text="Search"
                Command="{Binding DoSearch}"></Button>
    </Grid>
    <ListView Grid.Row="2" BackgroundColor="FloralWhite"
              VerticalOptions="Fill"
              HasUnevenRows="true"
              ItemsSource="{Binding Items}"
              IsGroupingEnabled="False"
              HorizontalOptions="Fill">
        <ListView.Behaviors>
            <behaviors:EventToCommandBehavior EventName="ItemSelected"
                                              Command="{Binding GoToDetails}"
                                              CommandParameter="{Binding Name}" />
        </ListView.Behaviors>
        <ListView.ItemTemplate>
```

Interface
Abstracting for platform specific features



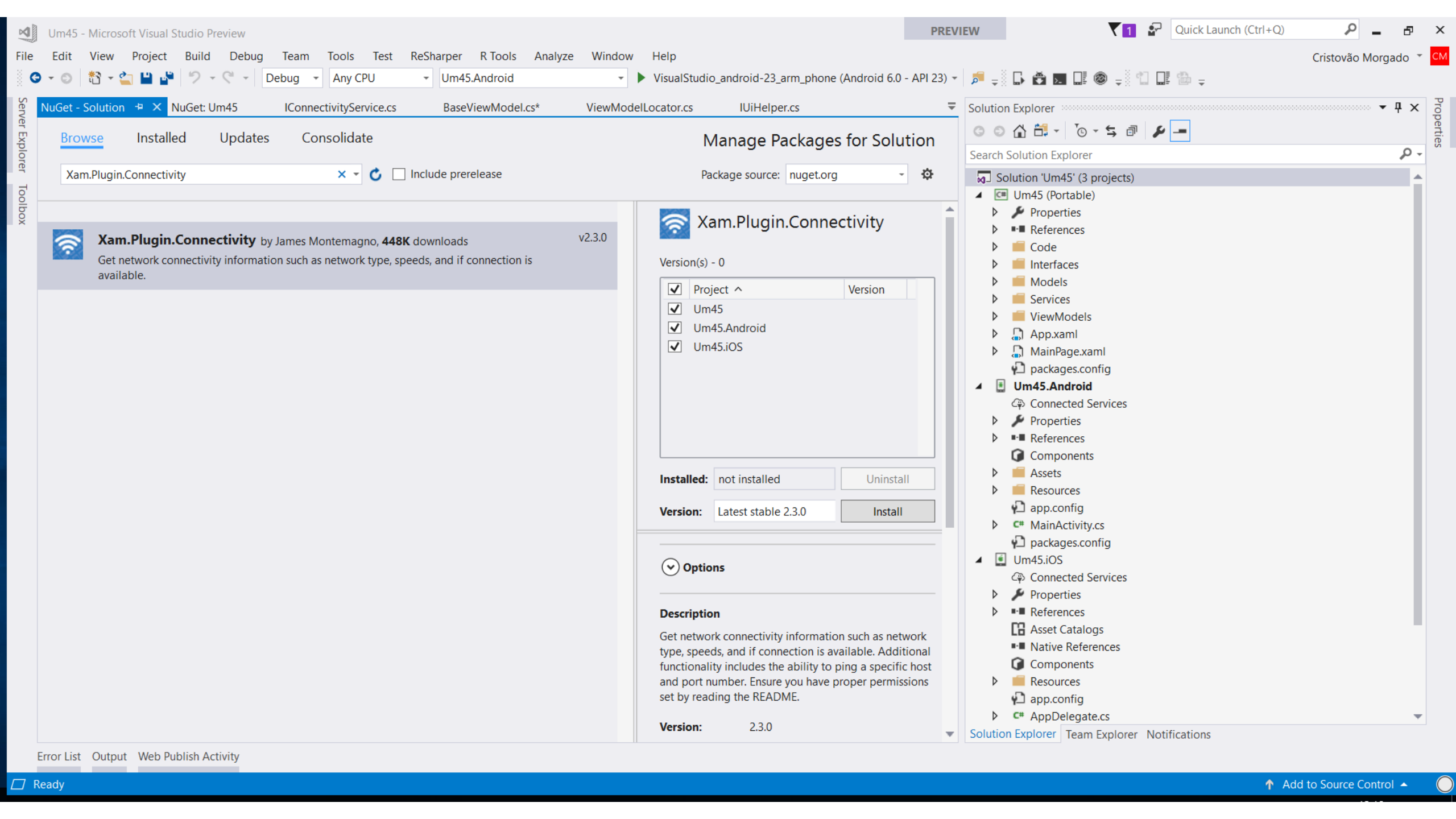




Platform specific services

```
namespace Um45.Droid
{
    [Activity(Label = "Um45", Icon = "@drawable/icon", Theme =
    0 references | Cristovão Morgado, 14 minutes ago | 1 author, 1 change
    public class MainActivity : global::Xamarin.Forms.Platform
    {
        0 references | Cristovão Morgado, 14 minutes ago | 1 author, 1 change
        protected override void OnCreate(Bundle bundle)
        {
            TabLayoutResource = Resource.Layout.Tabbar;
            ToolbarResource = Resource.Layout.Toolbar;

            base.OnCreate(bundle);
            Code.RegisterServices.Init();
            global::Xamarin.Forms.Forms.Init(this, bundle);
            LoadApplication(new Um45.App());
        }
    }
```

```
namespace Um45.Droid.Code
{
    1 reference | Cristovão Morgado, 14 minutes ago | 1 author, 1 change
    public static class RegisterServices
    {
        1 reference | Cristovão Morgado, 14 minutes ago | 1 author, 1 change
        public static void Init()
        {
            if (!SimpleIoc.Default.IsRegistered<ILogger>())
                SimpleIoc.Default.Register<ILogger>(() =>
                    new Um45.Droid.Services.Logger());


        }
    }
}
```


Um45 - Microsoft Visual Studio Preview
PREVIEW
Quick Launch (Ctrl+Q)
File Edit View Project Build Debug Team Tools Test ReSharper R Tools Analyze Window Help
Cristovão Morgado
Debug
Any CPU
Um45.Android
VisualStudio_android-23_arm_phone (Android 6.0 - API 23)
NuGet - Solution
NuGet: Um45
IConnectivityService.cs
BaseViewModel.cs*
ViewModelLocator.cs
IUiHelper.cs
Browse
Installed
Updates
Consolidate
Manage Packages for Solution
Xam.Plugin.Connectivity
Include prerelease
Package source: nuget.org
Xam.Plugin.Connectivity by James Montemagno, 448K downloads
v2.3.0
Get network connectivity information such as network type, speeds, and if connection is available.
Xam.Plugin.Connectivity
Version(s) - 0
Project
Version
Um45
Um45.Android
Um45.iOS
Installed: not installed
Uninstall
Version: Latest stable 2.3.0
Install
Options
Description
Get network connectivity information such as network type, speeds, and if connection is available. Additional functionality includes the ability to ping a specific host and port number. Ensure you have proper permissions set by reading the README.
Version: 2.3.0
Solution Explorer
Search Solution Explorer
Solution 'Um45' (3 projects)
Um45 (Portable)
Properties
References
Code
Interfaces
Models
Services
ViewModels
App.xaml
MainPage.xaml
packages.config
Um45.Android
Connected Services
Properties
References
Components
Assets
Resources
app.config
MainActivity.cs
packages.config
Um45.iOS
Connected Services
Properties
References
Asset Catalogs
Native References
Components
Resources
app.config
AppDelegate.cs
Solution Explorer
Team Explorer
Notifications
Server Explorer
Toolbox
Properties
Error List
Output
Web Publish Activity
Ready
Add to Source Control

Converters

```
namespace Um45.Controls.Converters
{
    0 references | Cristovão Morgado, 15 minutes ago | 1 author, 1 change
    public class BooleanConverter : IValueConverter
    {
        1 reference | Cristovão Morgado, 15 minutes ago | 1 author, 1 change
        public object Convert(object value, Type targetType, object parameter, CultureInfo culture)
        {
            var result = false;
            try
            {
                result = (bool) value;
            }
            catch
            {
                //ignored
            }
            return !result;
        }

        1 reference | Cristovão Morgado, 15 minutes ago | 1 author, 1 change
        public object ConvertBack(object value, Type targetType, object parameter, CultureInfo culture)
        {
            return null;
        }
    }
}
```

Behaviors

```
public class MinLengthBehavior : Behavior<Entry>
{

    5 references | 0 changes | 0 authors, 0 changes
    protected override void OnAttachedTo(Entry bindable)
    {
        bindable.TextChanged += OnTextChanged;

        base.OnAttachedTo(bindable);
    }

    2 references | 0 changes | 0 authors, 0 changes
    void OnTextChanged(object sender, TextChangedEventArgs args)
    {
        if (args.NewTextValue == null) return;

        var isValid = args.NewTextValue.Length > 3;

        ((Entry)sender).TextColor = isValid ? Color.Default : Color.Red;
    }

    5 references | 0 changes | 0 authors, 0 changes
    protected override void OnDetachingFrom(Entry bindable)
    {
        bindable.TextChanged -= OnTextChanged;

        base.OnDetachingFrom(bindable);
    }
}
```

Behaviors

```
    <Entry  Margin="10" HorizontalOptions="FillAndExpand" BackgroundColor="White"
            Text="{Binding SearchQuery, Mode=TwoWay}">
        <Entry.Behaviors>
            <behaviors:MinLengthBehavior >

            </behaviors:MinLengthBehavior>

        </Entry.Behaviors>

    </Entry>
```

The ViewModelLocator

```
using GalaSoft.MvvmLight.Ioc;
using Microsoft.Practices.ServiceLocation;

namespace Um45.ViewModels
{
    4 references | Cristovão Morgado, 11 minutes ago | 1 author, 1 change
    public class ViewModelLocator
    {
        1 reference | Cristovão Morgado, 11 minutes ago | 1 author, 1 change
        public ViewModelLocator()
        {
            ServiceLocator.SetLocatorProvider(() => SimpleIoc.Default);
            SimpleIoc.Default.Register<HomeViewModel>();
            SimpleIoc.Default.Register<DetailsViewModel>();
        }

        1 reference | Cristovão Morgado, 11 minutes ago | 1 author, 1 change
        public HomeViewModel Home => ServiceLocator.Current.GetInstance<HomeViewModel>();

        0 references | Cristovão Morgado, 11 minutes ago | 1 author, 1 change
        public DetailsViewModel Details => ServiceLocator.Current.GetInstance<DetailsViewModel>();
    }
}
```

Using the ViewModelLocator

```
namespace Um45
{
    6 references | Cristovão Morgado, 40 minutes ago | 1 author, 1 change
    public partial class App : Application
    {
        private static ViewModelLocator _locator;
            1 reference | Cristovão Morgado, 40 minutes ago | 1 author, 1 change
        public static ViewModelLocator Locator => _locator ?? (_locator = new ViewModelLocator());
```

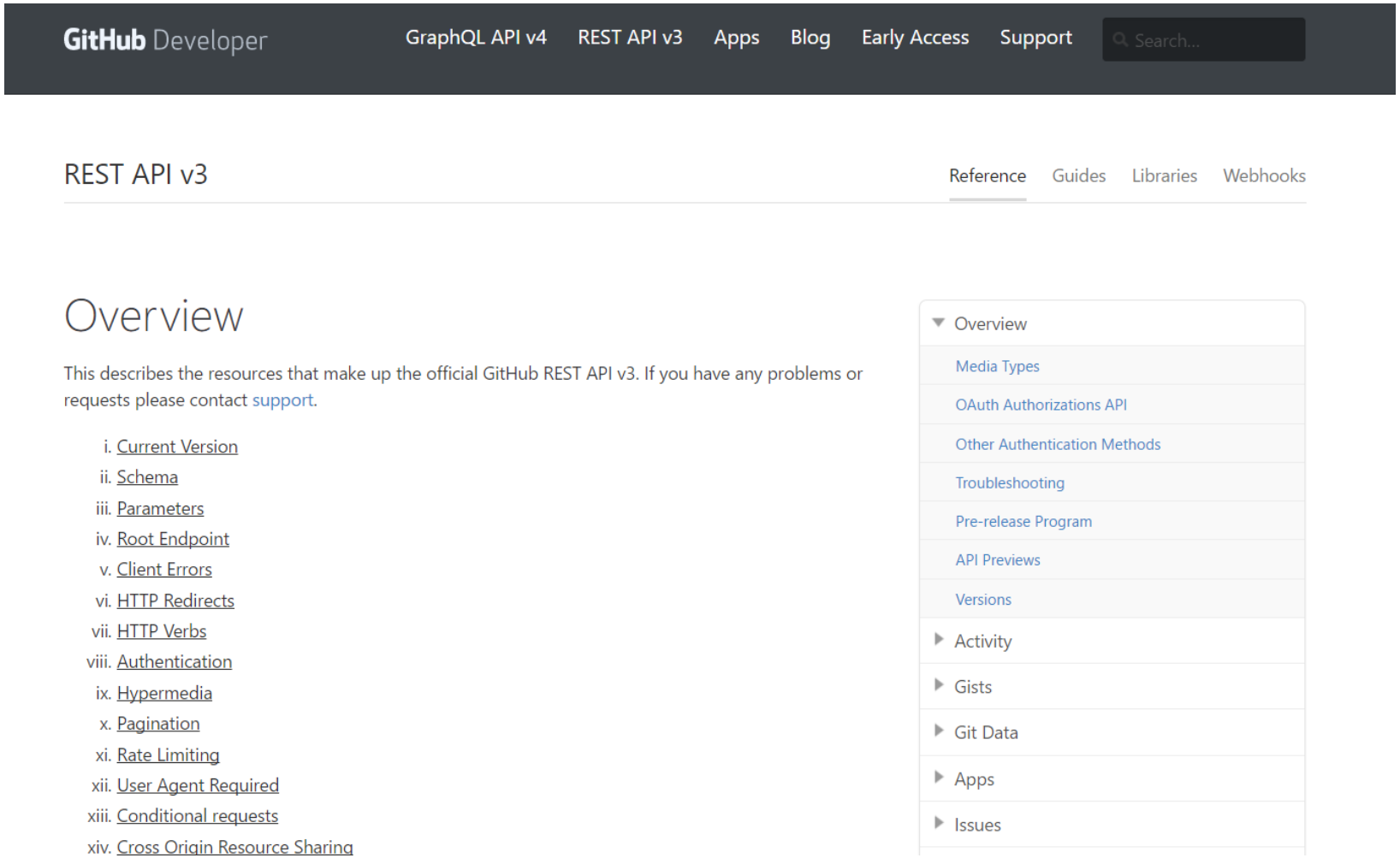
```
namespace Um45.Views
{
    5 references | Cristovão Morgado, 40 minutes ago | 1 author, 1 change
    public partial class MainPage : ContentPage
    {
        1 reference | Cristovão Morgado, 40 minutes ago | 1 author, 1 change
        public MainPage()
        {
            InitializeComponent();

            BindingContext = App.Locator.Home;
        }
    }
}
```

Dependency Injection – Boot Forms (App.cs)

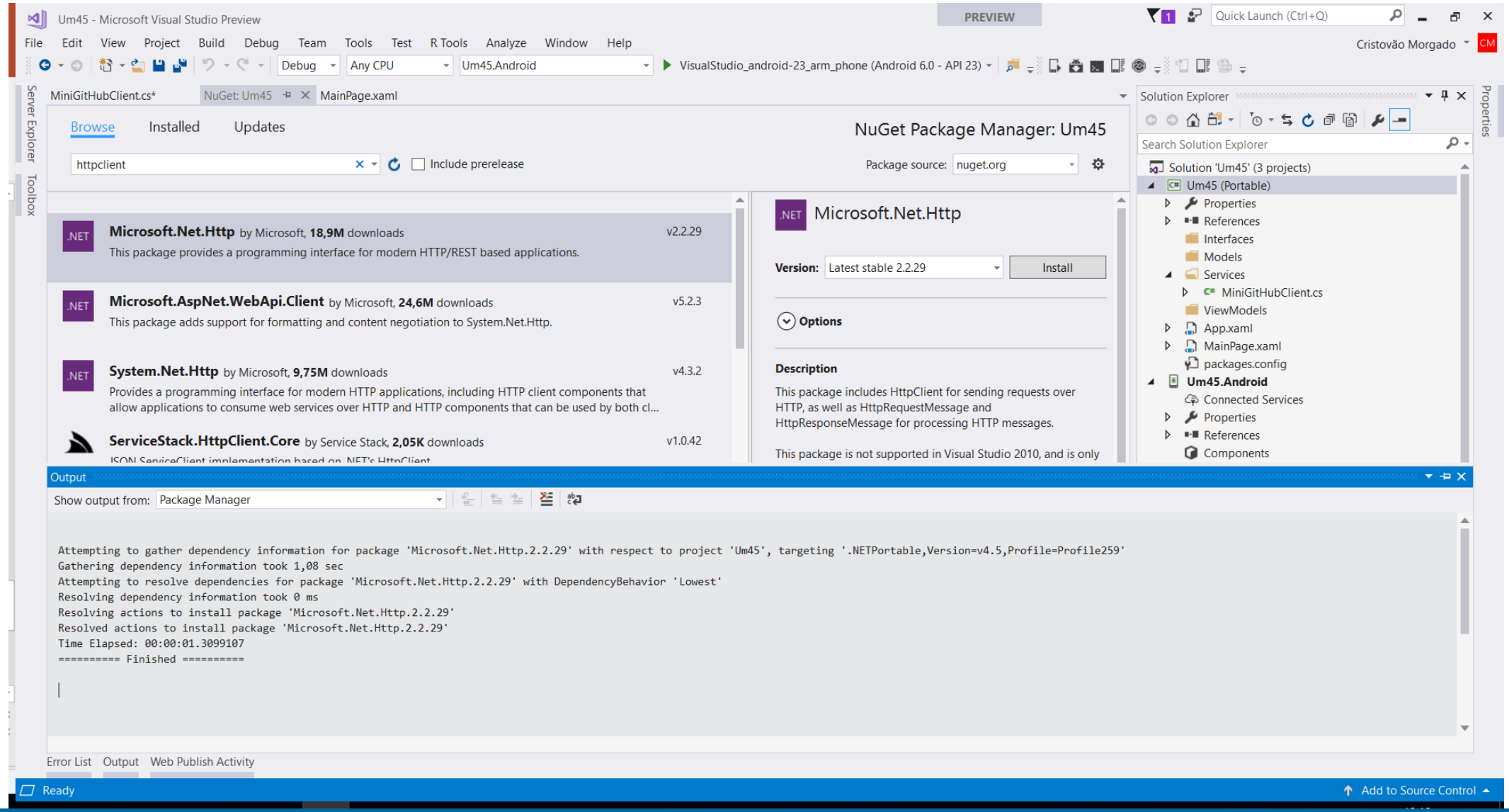
```
namespace Um45
{
    4 references | Cristovão Morgado, 12 minutes ago | 1 author, 1 change
    public partial class App : Application
    {
        private static ViewModelLocator _locator;
        1 reference | Cristovão Morgado, 12 minutes ago | 1 author, 1 change
        public static ViewModelLocator Locator => _locator ?? (_locator = new ViewModelLocator());

        1 reference | Cristovão Morgado, 12 minutes ago | 1 author, 1 change
        public static Page Page { get; private set; }
        2 references | Cristovão Morgado, 12 minutes ago | 1 author, 1 change
        public App()
        {
            InitializeComponent();

            if (!SimpleIoc.Default.IsRegistered<INavigationService>())
                SimpleIoc.Default.Register<INavigationService>(() => new Um45.Services.UberNavigation());

            if (!SimpleIoc.Default.IsRegistered<IUiHelper>())
                SimpleIoc.Default.Register<IUiHelper>(() => new Um45.Services.UiHelper());

            if (!SimpleIoc.Default.IsRegistered<IConnectivityService>())
                SimpleIoc.Default.Register<IConnectivityService>(() => new Um45.Services.ConnectivityService());

            MainPage = new MainPage();
            Page = MainPage;
        }
```

Let's get data from GitHub API

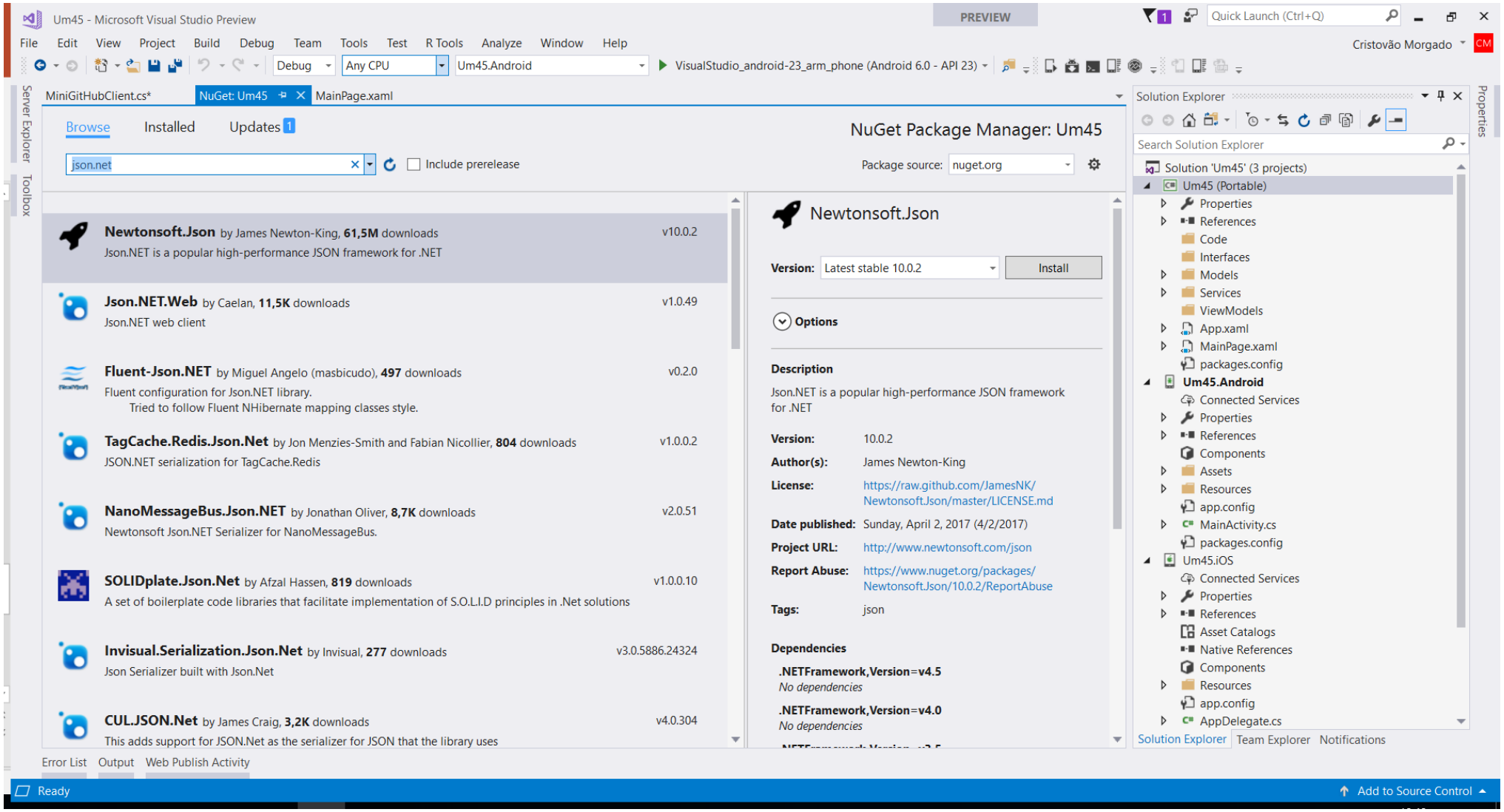
https://developer.github.com/v3/



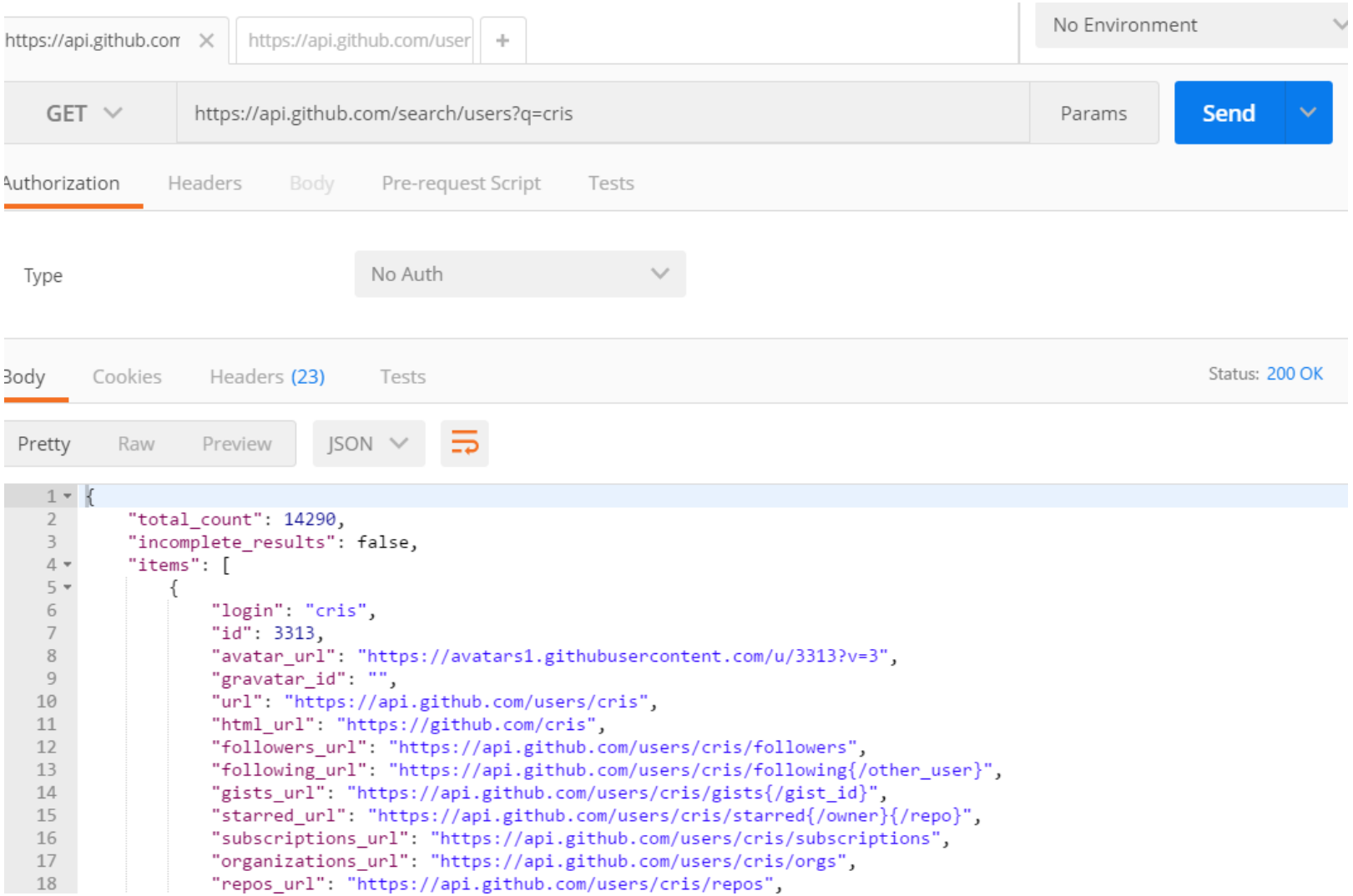
We need a HTTP Client



We need some way to deal with JSON: JSON.Net



We have JSON, now we have C# POCO's



We have JSON, now we have C# POCO's

```
namespace Um45.Models.Rest
{
    2 references | Cristovão Morgado, 47 minutes ago | 1 author, 1 change
    public class SearchResults
    {
        [JsonProperty(PropertyName = "total_count")]
        1 reference | Cristovão Morgado, 47 minutes ago | 1 author, 1 change
        public int TotalCount { get; set; }

        [JsonProperty(PropertyName = "incomplete_results")]
        0 references | Cristovão Morgado, 47 minutes ago | 1 author, 1 change
        public bool IncompleteResults { get; set; }

        [JsonProperty(PropertyName = "items")]
        1 reference | Cristovão Morgado, 47 minutes ago | 1 author, 1 change
        public Item[] Items { get; set; }
    }
}
```

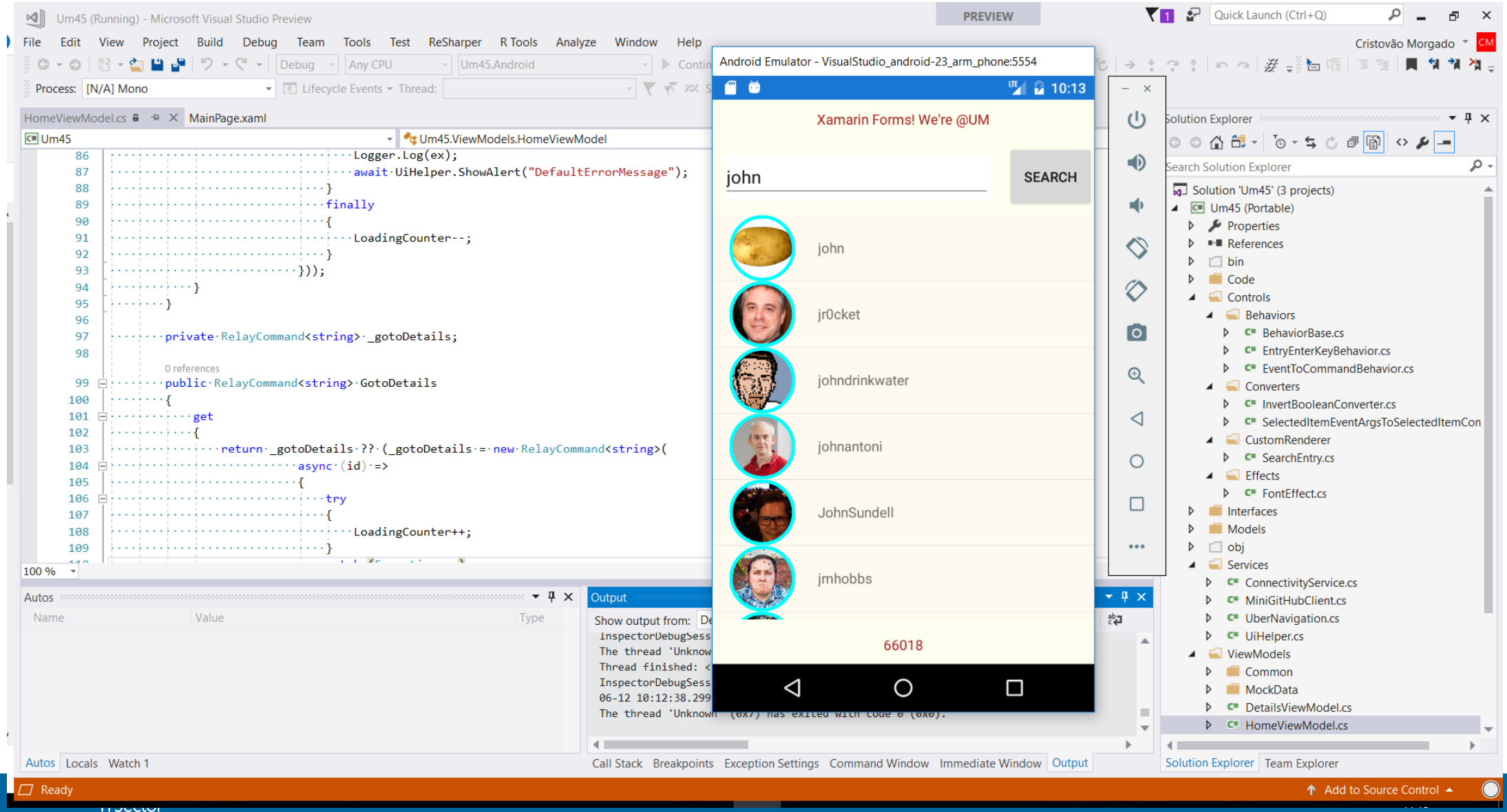
```
using Newtonsoft.Json;

namespace Um45.Models.Rest
{
    1 reference | Cristovão Morgado, 45 minutes ago | 1 author, 1 change
    public class Item
    {
        [JsonProperty(PropertyName = "login")]
        1 reference | Cristovão Morgado, 45 minutes ago | 1 author, 1 change
        public string Login { get; set; }

        [JsonProperty(PropertyName = "id")]
        0 references | Cristovão Morgado, 45 minutes ago | 1 author, 1 change
        public int Id { get; set; }

        [JsonProperty(PropertyName = "avatar_url")]
        1 reference | Cristovão Morgado, 45 minutes ago | 1 author, 1 change
        public string AvatarUrl { get; set; }

        [JsonProperty(PropertyName = "gravatar_id")]
        0 references | Cristovão Morgado, 45 minutes ago | 1 author, 1 change
        public string GravatarId { get; set; }

        [JsonProperty(PropertyName = "url")]
        0 references | Cristovão Morgado, 45 minutes ago | 1 author, 1 change
        public string Url { get; set; }

        [JsonProperty(PropertyName = "html_url")]
        0 references | Cristovão Morgado, 45 minutes ago | 1 author, 1 change
        public string HtmlUrl { get; set; }

        [JsonProperty(PropertyName = "followers_url")]
        0 references | Cristovão Morgado, 45 minutes ago | 1 author, 1 change
        public string FollowersUrl { get; set; }
```

A simple HTTP Client for github

```
public class MiniGitHubClient
{
    private readonly HttpClient _client = new HttpClient();

    1 reference
    public string BaseUrl { get; set; } = "https://api.github.com";

    1 reference
    public MiniGitHubClient()
    {
        _client.DefaultRequestHeaders.Add("User-Agent", "xamarin-um45");
    }

    1 reference
    public async Task<RestResult<SearchResults>> Search(string query)
    {
        return await DoHttpGet<SearchResults>($"/search/users?q={query}");
    }

    0 references
    public async Task<RestResult<UserDetails>> UserDetails(string login)
    {
        return await DoHttpGet<UserDetails>($"/users/{login}");
    }

    2 references
    private async Task<RestResult<T>> DoHttpGet<T>(string url)
    {
        var r = new RestResult<T>();
```

Now a walktrought

We have a running App!



Links

This presentation source code
https://github.com/cmorgado/Um45

Some of the best plugins
https://github.com/jamesmontemagno/Xamarin.Plugins

Get the tools
www.visualstudio.com

Nice Feed of development related news, tips, etc.
http://allaboutxamarin.com/
https://xamarinhelp.com/
http://www.Xamarin.com/

IT sector
Sistemas de Informação, SA

Q&A

@TheSaintr

CRISTOVAO.MORGADO@ITSECTOR.PT

IT sector
Sistemas de Informação, SA

Meet us @ our Stand

@TheSaintr

CRISTOVAO.MORGADO@ITSECTOR.PT

PORTO
Rua José Falcão, 151 - 1º/2º
4050-317 Porto
T: (+351) 222 058 272/3
F: (+351) 222 058 274

LISBOA
Praça Duque de Saldanha,
1 - 9º L/M
1050-094 Lisboa
(+351) 217 935 258
(+351) 217 958 334
lisboa@itsector.pt

BRAGA
Rua Padre António Vieira,
nº 105 R/C São Victor
4710-412 Braga
T: +351 253 137 501

ANGOLA
Condomínio Umbi Umbi
Morro Bento (Gamek) - Luanda
T: (+244) 930 237 770/1
F: (+244) 226 215 365

MOÇAMBIQUE
Rua Cahora Bassa, nº 230/231
1102 Maputo
T: (+258) 84 505 0575
T: (+258) 82 499 2508