**qwertyuiopasdfghjklzxcvbnmqwertyuiopasdfghjklzxcvbnmqwertyuiopasdfghjklzxcvbnmqwertyuiopasdfghjklzxcvbnmqwertyuiopasdfghjklzxcvbnmqwertyuiopasdfghjklzxcvbnmqwertyuiopasdfghjklzxcvbnmqwertyuiopasdfghjklzxcvbnmqwertyuiopasdfghjklzxcvbnmqwertyuiopasdfghjklzxcvbnmqwertyuiopasdfghjklzxcvbnmqwertyuiopasdfghjklzxcvbnmqwertyuiopasdfghjklzxcvbnmqwertyuiopasdfghjklzxcvbnmqwertyuiopasdfghjklzxcvbnmqwertyuiopasdfghjklzxcvbnmqwertyuiopasdfghjklzxcvbnmqwertyuiopasdfghjklzxcvbnmrtyuiopasdfghjklzxcvbnmqwertyuiopasdfghjklzxcvbnmqwertyuiopasdfghjklzxcvbnmqwertyuiopasdfghjklzxcvbnmqwertyuiopasdfghjklzxcvbnmqwertyuiopasdfghjklzxcvbnmqwertyuiopasdfghjklzxcvbnmqwertyuiopasdfghjklzxcvbnmqwertyuiopasdfghjklzxcvbnmqwertyuiopasdfghjklzxcvbnmqwertyuiopasdfghjklzxcvbnmqwertyuiopasdfghjklzxcvbnmqwertyuiopasdfghjklzxcvbnmrtyuiopasdfghjklzxcvbnmqwertyuiopasdfghjklzxcvbnmqwertyuiopasdfghjklzxcvbnmqwertyuiopasdfghjklzxcvbnmqwertyuiopasdfghjklzxcvbnmqwertyuiopasdfghjklzxcvbnmqwertyuiopasdfghjklzxcvbnmqwertyuiopasdfghjklzxcvbnmqwertyuiopasdfghjklzxcvbnmqwertyuiopasdfghjklzxcvbnmqwertyuiopasdfghjklzxcvbnmqwertyuiopasdfghjklzxcvbnmqwertyuiopasdfghjklzxcvbnmrtyuiopasdfghjklzxcvbnmqwertyuiopasdfghjklzxcvbnmqwertyuiopasdfghjklzxcvbnmqwertyuiopasdfghjklzxcvbnmqwertyuiopasdfghjklzxcvbnmqwertyuiopasdfghjklzxcvbnmqwertyuiopasdfghjklzxcvbnmqwertyuiopasdfghjklzxcvbnmqwertyuiopasdfghjklzxcvbnmqwertyuiopasdfghjklzxcvbnmqwertyuiopasdfghjklzxcvbnmqwertyuiopasdfghjklzxcvbnmqwertyuiopasdfghjklzxcvbnmrtyuiopasdfghjklzxcvbnmqwertyuiopasdfghjklzxcvbnmqwertyuiopasdfghjklzxcvbnmqwertyuiopasdfghjklzxcvbnmqwertyuiopasdfghjklzxcvbnmqwertyuiopasdfghjklzxcvbnmqwertyuiopasdfghjklzxcvbnmqwertyuiopasdfghjklzxcvbnmqwertyuiopasdfghjklzxcvbnmqwertyuiopasdfghjklzxcvbnmqwertyuiopasdfghjklzxcvbnmqwertyuiopasdfghjklzxcvbnmqwertyuiopasdfghjklzxcvbnmrtyuiopasdfghjklzxcvbnmqwertyuiopasdfghjklzxcvbnmqwertyuiopasdfghjklzxcvbnmqwertyuiopasdfghjklzxcvbnmqwertyuiopasdfghjklzxcvbnmqwertyuiopasdfghjklzxcvbnmqwertyuiopasdfghjklzxcvbnmqwertyuiopasdfghjklzxcvbnmqwertyuiopasdfghjklzxcvbnmqwertyuiopasdfghjklzxcvbnmqwertyuiopasdfghjklzxcvbnmqwertyuiopasdfghjklzxcvbnmqwertyuiopasdfghjklzxcvbnmrtyuiopasdfghjklzxcvbnmqwertyuiopasdfghjklzxcvbnmqwertyuiopasdfghjklzxcvbnmqwertyuiopasdfghjklzxcvbnmqwertyuiopasdfghjklzxcvbnmqwertyuiopasdfghjklzxcvbnmqwertyuiopasdfghjklzxcvbnmqwertyuiopasdfghjklzxcvbnmqwertyuiopasdfghjklzxcvbnmqwertyuiopasdfghjklzxcvbnmqwertyuiopasdfghjklzxcvbnmqwertyuiopasdfghjklzxcvbnmqwertyuiopasdfghjklzxcvbnmrtyuiopasdfghjklzxcvbnmqwertyuiopasdfghjklzxcvbnmqwertyuiopasdfghjklzxcvbnmqwertyuiopasdfghjklzxcvbnmqwertyuiopasdfghjklzxcvbnmqwertyuiopasdfghjklzxcvbnmqwertyuiopasdfghjklzxcvbnmqwertyuiopasdfghjklzxcvbnmqwertyuiopasdfghjklzxcvbnmqwertyuiopasdfghjklzxcvbnmqwertyuiopasdfghjklzxcvbnmqwertyuiopasdfghjklzxcvbnmqwertyuiopasdfghjklzxcvbnmqwwertyuiopasdfghjklzxcvbnmqwertyuiopasdfghjklzxcvbnmqwertyuiopasdfghjklzxcvbnmqwertyuiopasdfghjklzxcvbnm**

|  |
| --- |
| Technical Manual  MCSD Formative 1 PG2  9/7/2018  Leardo Taylor |

|  |  |
| --- | --- |
| Requirements | |
| Platform | Windows |
| Min Specs | 1 gig ram, 1ghz CPU, 200mg spaces |
| Requires Internet | True |

**Code**

using System;

using System.Collections.Generic;

using System.IO;

using System.Linq;

using System.Runtime.InteropServices.WindowsRuntime;

using Windows.Foundation;

using Windows.Foundation.Collections;

using Windows.UI.Xaml;

using Windows.UI.Xaml.Controls;

using Windows.UI.Xaml.Controls.Primitives;

using Windows.UI.Xaml.Data;

using Windows.UI.Xaml.Input;

using Windows.UI.Xaml.Media;

using Windows.UI.Xaml.Navigation;

using Windows.Devices.Geolocation;

using Windows.Services.Maps;

using Windows.UI.Xaml.Controls.Maps;

using Windows.Storage.Streams;

using MyToolkit.Multimedia;

// The Blank Page item template is documented at https://go.microsoft.com/fwlink/?LinkId=402352&clcid=0x409

namespace CTU\_Minimap

{

/// <summary>

/// An empty page that can be used on its own or navigated to within a Frame.

/// </summary>

public sealed partial class MainPage : Page

{

bool? isVidPaused = null;

public MainPage()

{

this.InitializeComponent();

HideUI();

HideControls();

}

#region Buttons

private void btnBegin\_Click(object sender, RoutedEventArgs e)

{

btnBegin.Content = "Settings";

MainMap.Visibility = Visibility.Visible;

HideCourses();

}

private void btnPEC\_Tapped(object sender, TappedRoutedEventArgs e)

{

AdjustUI("PortElizabeth", "Q4KXr3BjQFw");

ShowControls();

ShowCourses();

}

private void btnCTC\_Tapped(object sender, TappedRoutedEventArgs e)

{

AdjustUI("CapeTown", "1H9l0K70114");

ShowControls();

ShowCourses();

}

private void btnPRC\_Tapped(object sender, TappedRoutedEventArgs e)

{

AdjustUI("Pretoria", "YPocrRnG638");

ShowControls();

ShowCourses();

}

private void btnCourse1\_Tapped(object sender, TappedRoutedEventArgs e)

{

AdjustUI("MCSD");

HideCourses();

}

private void btnCourse2\_Tapped(object sender, TappedRoutedEventArgs e)

{

AdjustUI("MCSA");

HideCourses();

}

private void btnCourse3\_Tapped(object sender, TappedRoutedEventArgs e)

{

AdjustUI("WEB1");

HideCourses();

}

private void btnCourse4\_Tapped(object sender, TappedRoutedEventArgs e)

{

AdjustUI("GD1");

HideCourses();

}

private void btnToggleMap\_Tapped(object sender, TappedRoutedEventArgs e)

{

HideCourses();

if (btnToggleMap.IsChecked)

{

vidOutput.Visibility = Visibility.Collapsed;

MainMap.Visibility = Visibility.Visible;

btnPausePlay.Visibility = Visibility.Collapsed;

vidOutput.Pause();

isVidPaused = true;

}

else

{

MainMap.Visibility = Visibility.Collapsed;

vidOutput.Visibility = Visibility.Visible;

btnPausePlay.Visibility = Visibility.Visible;

vidOutput.Play();

isVidPaused = false;

btnPausePlay.Text = "Pause Video";

}

}

private void btnPausePlay\_Tapped(object sender, TappedRoutedEventArgs e)

{

if (isVidPaused == true)

{

vidOutput.Play();

isVidPaused = false;

btnPausePlay.Text = "Pause Video";

}

else

{

vidOutput.Pause();

isVidPaused = true;

btnPausePlay.Text = "Play Video";

}

}

#endregion

#region Methods

private void HideUI()

{

MainMap.Visibility = Visibility.Collapsed;

vidOutput.Visibility = Visibility.Collapsed;

txtLocation.Visibility = Visibility.Collapsed;

txtContact.Visibility = Visibility.Collapsed;

txtCourse.Visibility = Visibility.Collapsed;

}

private void HideControls()

{

ControlTitle.Visibility = Visibility.Collapsed;

ControlSeperator.Visibility = Visibility.Collapsed;

btnToggleMap.Visibility = Visibility.Collapsed;

btnPausePlay.Visibility = Visibility.Collapsed;

}

private void HideCourses()

{

CourseTitle1.Visibility = Visibility.Collapsed;

CourseTitle2.Visibility = Visibility.Collapsed;

CourseTitle3.Visibility = Visibility.Collapsed;

CourseSeperator1.Visibility = Visibility.Collapsed;

CourseSeperator2.Visibility = Visibility.Collapsed;

CourseSeperator3.Visibility = Visibility.Collapsed;

btnCourse1.Visibility = Visibility.Collapsed;

btnCourse2.Visibility = Visibility.Collapsed;

btnCourse3.Visibility = Visibility.Collapsed;

btnCourse4.Visibility = Visibility.Collapsed;

btnCourse5.Visibility = Visibility.Collapsed;

btnCourse6.Visibility = Visibility.Collapsed;

btnCourse7.Visibility = Visibility.Collapsed;

btnCourse8.Visibility = Visibility.Collapsed;

btnCourse9.Visibility = Visibility.Collapsed;

btnCourse10.Visibility = Visibility.Collapsed;

btnCourse11.Visibility = Visibility.Collapsed;

btnCourse12.Visibility = Visibility.Collapsed;

}

private void ShowCourses()

{

CourseTitle1.Visibility = Visibility.Visible;

CourseTitle2.Visibility = Visibility.Visible;

CourseTitle3.Visibility = Visibility.Visible;

CourseSeperator1.Visibility = Visibility.Visible;

CourseSeperator2.Visibility = Visibility.Visible;

CourseSeperator3.Visibility = Visibility.Visible;

btnCourse1.Visibility = Visibility.Visible;

btnCourse2.Visibility = Visibility.Visible;

btnCourse3.Visibility = Visibility.Visible;

btnCourse4.Visibility = Visibility.Visible;

btnCourse5.Visibility = Visibility.Visible;

btnCourse6.Visibility = Visibility.Visible;

btnCourse7.Visibility = Visibility.Visible;

btnCourse8.Visibility = Visibility.Visible;

btnCourse9.Visibility = Visibility.Visible;

btnCourse10.Visibility = Visibility.Visible;

btnCourse11.Visibility = Visibility.Visible;

btnCourse12.Visibility = Visibility.Visible;

btnCourse1.Text = "MCSD";

btnCourse2.Text = "MCSA";

btnCourse3.Text = "WEB1";

btnCourse4.Text = "GD1";

btnCourse5.Text = "MCSD";

btnCourse6.Text = "MCSA";

btnCourse7.Text = "WEB1";

btnCourse8.Text = "GD1";

btnCourse9.Text = "MCSD";

btnCourse10.Text = "MCSA";

btnCourse11.Text = "WEB1";

btnCourse12.Text = "GD1";

}

private void ShowControls()

{

ControlTitle.Visibility = Visibility.Visible;

ControlSeperator.Visibility = Visibility.Visible;

btnToggleMap.Visibility = Visibility.Visible;

btnPausePlay.Visibility = Visibility.Visible;

}

private void AdjustUI(string Campus, string YouTubeId)

{

GoToLocation(Campus);

btnToggleMap.IsChecked = false;

isVidPaused = false;

btnPausePlay.Text = "Pause Video";

PlayVid(YouTubeId);

MainMap.Visibility = Visibility.Collapsed;

vidOutput.Visibility = Visibility.Visible;

txtLocation.Visibility = Visibility.Visible;

txtLocation.Text = File.ReadAllText("resources/Campus\_Location/"+ Campus +".txt");

txtContact.Visibility = Visibility.Visible;

txtContact.Text = File.ReadAllText("resources/Contact/" + Campus + ".txt");

txtCourse.Visibility = Visibility.Visible;

txtCourse.Text = File.ReadAllText("resources/Campus\_Descriptions/" + Campus + ".txt");

}

private void AdjustUI(string Course)

{

txtCourse.Text = File.ReadAllText("resources/Course\_Descriptions/" + Course + ".txt");

}

private async void PlayVid(string YouTubeId)

{

try

{

YouTubeUri url = await YouTube.GetVideoUriAsync(YouTubeId, YouTubeQuality.Quality360P);

vidOutput.Source = url.Uri;

if(isVidPaused == null) vidOutput.Play();

btnToggleMap.IsEnabled = true;

}

catch (Exception)

{

// TODO show error (video uri not found)

}

}

private async void GoToLocation(string location)

{

switch (location)

{

case ("PortElizabeth"):

BasicGeoposition geopositionPE = new BasicGeoposition();

geopositionPE.Longitude = 25.570099;

geopositionPE.Latitude = -33.94507;

Geopoint myPointPE = new Geopoint(geopositionPE);

MainMap.Center = myPointPE;

MapScene mapScenePE = MapScene.CreateFromLocationAndRadius(new Geopoint(geopositionPE), 500, 170, 50);

await MainMap.TrySetSceneAsync(mapScenePE);

break;

case ("CapeTown"):

BasicGeoposition geopositionCT = new BasicGeoposition();

geopositionCT.Longitude = 18.465496;

geopositionCT.Latitude = -33.979565;

Geopoint myPointCT = new Geopoint(geopositionCT);

MainMap.Center = myPointCT;

MapScene mapSceneCT = MapScene.CreateFromLocationAndRadius(new Geopoint(geopositionCT), 500, 170, 50);

await MainMap.TrySetSceneAsync(mapSceneCT);

break;

case ("Pretoria"):

BasicGeoposition geopositionPR = new BasicGeoposition();

geopositionPR.Longitude = 28.279011;

geopositionPR.Latitude = -25.783079;

Geopoint myPointPR = new Geopoint(geopositionPR);

MainMap.Center = myPointPR;

MapScene mapScenePR = MapScene.CreateFromLocationAndRadius(new Geopoint(geopositionPR), 500, 170, 50);

await MainMap.TrySetSceneAsync(mapScenePR);

break;

}

}

private async void MainMap\_Loaded(object sender, RoutedEventArgs e)

{

if (MainMap.Is3DSupported)

{

MainMap.Style = MapStyle.Aerial3DWithRoads;

MainMap.MapServiceToken = "";

BasicGeoposition geopositionPE = new BasicGeoposition();

geopositionPE.Longitude = 25.570099;

geopositionPE.Latitude = -33.94507;

Geopoint myPointPE = new Geopoint(geopositionPE);

MapIcon myPOIPE = new MapIcon { Location = myPointPE, Title = "Port Elizabeth Campus" };

myPOIPE.Image = RandomAccessStreamReference.CreateFromUri(new Uri("ms-appx:///Assets/pin.png"));

MainMap.MapElements.Add(myPOIPE);

BasicGeoposition geopositionCT = new BasicGeoposition();

geopositionCT.Longitude = 18.465496;

geopositionCT.Latitude = -33.979565;

Geopoint myPointCT = new Geopoint(geopositionCT);

MapIcon myPOICT = new MapIcon { Location = myPointCT, Title = "Cape Town" };

myPOICT.Image = RandomAccessStreamReference.CreateFromUri(new Uri("ms-appx:///Assets/pin.png"));

MainMap.MapElements.Add(myPOICT);

BasicGeoposition geopositionPR = new BasicGeoposition();

geopositionPR.Longitude = 28.279011;

geopositionPR.Latitude = -25.783079;

Geopoint myPointPR = new Geopoint(geopositionPR);

MapIcon myPOIPR = new MapIcon { Location = myPointPR, Title = "Pretoria" };

myPOIPR.Image = RandomAccessStreamReference.CreateFromUri(new Uri("ms-appx:///Assets/pin.png"));

MainMap.MapElements.Add(myPOIPR);

BasicGeoposition geopositionSA = new BasicGeoposition();

geopositionSA.Longitude = 22.937506;

geopositionSA.Latitude = -30.559482;

Geopoint myPointSA = new Geopoint(geopositionSA);

MainMap.Center = myPointSA;

}

}

#endregion

}

}