**ПРАВИТЕЛЬСТВО РОССИЙСКОЙ ФЕДЕРАЦИИ**

**НАЦИОНАЛЬНЫЙ ИССЛЕДОВАТЕЛЬСКИЙ УНИВЕРСИТЕТ**

**«ВЫСШАЯ ШКОЛА ЭКОНОМИКИ»**

Факультет компьютерных наук

Департамент программной инженерии

|  |  |
| --- | --- |
| СОГЛАСОВАНО  Руководитель департамента программной инженерии факультета компьютерных наук  \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ С. М. Авдошин  «\_\_\_» \_\_\_\_\_\_\_\_\_\_\_\_\_ 2019 г. | УТВЕРЖДАЮ  Академический руководитель образовательной программы «Программная инженерия»  \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ В.В. Шилов  «\_\_\_» \_\_\_\_\_\_\_\_\_\_\_\_\_ 2019 г. |

|  |  |
| --- | --- |
| *Подп. и дата* |  |
| *Инв. № дубл.* |  |
| *Взам. инв. №* |  |
| *Подп. и дата* |  |
| *Инв. № подл* | *RU.17701729.04.01-01 12 01-1* |

**ПРОГРАММНЫЙ СИНТЕЗАТОР В ФОРМАТЕ ИНСТРУМЕНТА VST**

**Текст программы**

**ЛИСТ УТВЕРЖДЕНИЯ**

**RU.17701729.04.01-01 12 01-1-ЛУ**

|  |  |
| --- | --- |
|  | Исполнитель:  студент группы БПИ182  \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ /М. А. Шакура /  «\_\_\_\_»\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ 2019 г. |

**2019**

|  |
| --- |
| УТВЕРЖДЕН  RU.17701729.04.01-01 12 01-1-ЛУ |

|  |  |
| --- | --- |
| *Подп. и дата* |  |
| *Инв. № дубл.* |  |
| *Взам. инв. №* |  |
| *Подп. и дата* |  |
| *Инв. № подл* | *RU.17701729.04.01-01 12 01-1* |

**ПРОГРАММНЫЙ СИНТЕЗАТОР В ФОРМАТЕ ИНСТРУМЕНТА VST**

**Текст программы**

**RU.17701729.04.01-01 12 01-1**

**Листов 183**

**2019**

# АННОТАЦИЯ

В данном программном документе представлен текст программы Приложение активности и уведомления ВШЭ». В разделе «Текст программы» находится текст программы, распределённый по файлам.

**СОДЕРЖАНИЕ**

[1. ТЕКСТ ПРОГРАММЫ 5](#_Toc8772577)

[1.1. Components\AudioComponent.cs 5](#_Toc8772578)

[1.2. Components\AudioComponentWithParameters.cs 5](#_Toc8772579)

[1.3. Components\DelayManager.cs 7](#_Toc8772580)

[1.4. Components\DistortionManager.cs 16](#_Toc8772581)

[1.5. Envelope.cs 24](#_Toc8772582)

[1.6. Components\EnvelopesManager.cs 28](#_Toc8772583)

[1.7. Components\Filter.cs 35](#_Toc8772584)

[1.8. Components\FiltersManager.cs 38](#_Toc8772585)

[1.9. Components\Oscillator.cs 43](#_Toc8772586)

[1.10. Components\OscillatorsManager.cs 45](#_Toc8772587)

[1.11. Components\ParameterFilter.cs 49](#_Toc8772588)

[1.12. Components\Routing.cs 52](#_Toc8772589)

[1.13. Components\Voice.cs 56](#_Toc8772590)

[1.14. Components\VoicesManager.cs 62](#_Toc8772591)

[1.15. DSP\Effects\Delay\DelayLine.cs 68](#_Toc8772592)

[1.16. DSP\Effects\Delay\IDelay.cs 71](#_Toc8772593)

[1.17. DSP\Effects\Delay\PingPongDelay.cs 72](#_Toc8772594)

[1.18. DSP\Effects\Delay\StereoOffsetDelay.cs 74](#_Toc8772595)

[1.19. DSP\Effects\Delay\VariousTimeDelay.cs 77](#_Toc8772596)

[1.20. DSP\Effects\Distortion\AbsClipper.cs 79](#_Toc8772597)

[1.21. DSP\Effects\Distortion\BitCrusher.cs 80](#_Toc8772598)

[1.22. DSP\Effects\Distortion\CubicClipper.cs 81](#_Toc8772599)

[1.23. DSP\Effects\Distortion\IDistortion.cs 82](#_Toc8772600)

[1.24. DSP\Effects\Distortion\SampleRateReduction.cs 82](#_Toc8772601)

[1.25. DSP\Effects\Distortion\SoftClipper.cs 84](#_Toc8772602)

[1.26. DSP\Filters\DCBlocker.cs 85](#_Toc8772603)

[1.27. DSP\Filters\Downsampler.cs 87](#_Toc8772604)

[1.28. DSP\Filters\SvfFilter.cs 97](#_Toc8772605)

[1.29. DSP\Generators\AdsrEnvelope.cs 103](#_Toc8772606)

[1.30. DSP\Generators\SineLFO.cs 110](#_Toc8772607)

[1.31. DSP\Generators\WaveTableOscillators.cs 111](#_Toc8772608)

[1.32. Plugin Parts\AudioProcessor.cs 117](#_Toc8772609)

[1.33. Plugin Parts\MidiProcessor.cs 119](#_Toc8772610)

[1.34. Plugin Parts\ParameterFactory.cs 122](#_Toc8772611)

[1.35. Plugin Parts\Plugin.cs 124](#_Toc8772612)

[1.36. Plugin Parts\PluginCommandStub.cs 127](#_Toc8772613)

[1.37. Plugin Parts\PluginEditor.cs 127](#_Toc8772614)

[1.38. Plugin Parts\PluginPrograms.cs 131](#_Toc8772615)

[1.39. UI\BipolarKnob.cs 132](#_Toc8772616)

[1.40. UI\EditorView.xaml 133](#_Toc8772617)

[1.41. UI\EditorView.xaml.cs 150](#_Toc8772618)

[1.42. UI\Knob.xaml 159](#_Toc8772619)

[1.43. UI\Knob.xaml.cs 161](#_Toc8772620)

[1.44. Converters.cs 168](#_Toc8772621)

[1.45. Utilities.cs 178](#_Toc8772622)

# ТЕКСТ ПРОГРАММЫ

## Models\AppData.cs

using System;

using System.Collections.Generic;

using Xamarin.Forms;

using System.IO;

using Newtonsoft.Json;

namespace App5.Models

{

/// <summary>

/// Sctruct of Data which writes to files

/// </summary>

struct FileStruct

{

public bool isrus { set; get; }

public bool IsThemeWhite { set; get; }

public List<string> Links;

}

static class AppData

{

static public Action SettingChanged;

static public List<string> Links = new List<string>();

/// <summary>

/// Langs

/// </summary>

static public string ru;

static public string en;

static public bool isrus { set; get; } = true;

/// <summary>

/// Colors

/// </summary>

static public bool IsThemeWhite { set; get; } = true;

static public Color BackgroundColor { get => IsThemeWhite ? Color.White : Color.FromRgb(16, 16, 16); }

static public Color BarBackgroundColor { get => IsThemeWhite ? Color.White : Color.FromRgb(10, 10, 10); }

static public Color FrontColor { get => IsThemeWhite ? Color.Black : Color.White; }

static AppData()

{

SettingChanged += Load;

/// <summary>

/// Read from file

/// </summary>

try

{

FileStruct a;

string json;

string path = Environment.GetFolderPath(Environment.SpecialFolder.Personal);

string filePath = Path.Combine(path, "AppData.txt");

using (var file = File.Open(filePath, FileMode.Open, FileAccess.Read))

using (var strm = new StreamReader(file))

{

json = strm.ReadToEnd();

}

a = JsonConvert.DeserializeObject<FileStruct>(json);

isrus = a.isrus;

IsThemeWhite = a.IsThemeWhite;

Links = a.Links;

}

catch (Exception e){}

}

/// <summary>

/// Write to file

/// </summary>

static void Load()

{

try

{

FileStruct a = new FileStruct() {isrus=isrus,

IsThemeWhite= IsThemeWhite,

Links = Links};

string json = JsonConvert.SerializeObject(a);

string path = Environment.GetFolderPath(Environment.SpecialFolder.Personal);

string filePath = Path.Combine(path, "AppData.txt");

using (var file = File.Open(filePath, FileMode.Create, FileAccess.Write))

using (var strm = new StreamWriter(file))

{

strm.Write(json);

}

}

catch (Exception e){}

}

}

}

## Models\Item.cs

using Xamarin.Forms;

namespace App5.Models

{

public class Item

{

public string Type { get; set; }

public string Day { get; set; } = "";

public string Header { get; set; } = "";

public string Description { get => string.Join(", ", Tags); }

public string ExtraString { get => string.Join(", ", Extra); }

public string Link { get; set; } = "";

public string Place { get; set; } = "";

public string Time { get; set; } = "";

public string Date { get => Time == "" ? Day : Day + ", " + Time; }

public string[] Tags { set; get; } = { };

public string[] Extra { set; get; } = { };

public bool DayVisible { get => Type == "Day"; }

public bool ExtraVisible { get => ExtraString != "" && Type == "Event"; }

public bool DescriptionVisible { get => Description != "" && Type == "Event"; }

public bool TimeVisible { get => Time != "" && Type == "Event"; }

public bool PlaceVisible { get => Place != "" && Type == "Event"; }

public bool TagsVisible { get => Tags.Length > 0 && Type == "Event"; }

public bool HeaderVisible { get => Type == "Event"; }

public bool Clickable { get => Type == "Event"; }

public string FavSource { get => AppData.Links.Contains(Link) ? "Added.png" : "fplus.png"; }

public Color FrontColor { get => AppData.FrontColor;}

public Color BackgroundColor { get => AppData.BackgroundColor; }

}

}

## Servises\IDataStore.cs

using System.Collections.Generic;

using System.Threading.Tasks;

namespace App5.Services

{

public interface IDataStore<T>

{

Task<IEnumerable<T>> GetItemsAsync(bool forceRefresh = false);

}

}

## Servises\MockDataStore.cs

using System.Collections.Generic;

using System.IO;

using System.Linq;

using System.Net;

using System.Threading.Tasks;

using App5.Models;

using Newtonsoft.Json;

using App5.Views;

namespace App5.Services

{

public class MockDataStore : IDataStore<Item>

{

/// <summary>

/// List of Items

/// </summary>

List<Item> items;

/// <summary>

/// Request to server

/// </summary>

/// <param name="url">Link</param>

/// <returns></returns>

static public string Get(string url)

{

HttpWebRequest request = (HttpWebRequest)WebRequest.Create(url);

request.AutomaticDecompression = DecompressionMethods.GZip | DecompressionMethods.Deflate;

try

{

using (HttpWebResponse response = (HttpWebResponse)request.GetResponse())

using (Stream stream = response.GetResponseStream())

using (StreamReader reader = new StreamReader(stream))

return reader.ReadToEnd();

}

catch

{

return null;

}

}

/// <summary>

/// Reloadl Data from server

/// </summary>

static public void RealoadData()

{

AppData.ru = Get("https://shakura.dev/hseapi");

AppData.en = Get("https://shakura.dev/hseapien");

AboutPage.f();

}

/// <summary>

/// Local relaod data after swiching language

/// </summary>

public MockDataStore()

{

Item[] data = JsonConvert.DeserializeObject<Item[]>(AppData.isrus ? AppData.ru : AppData.en);

items = data.OfType<Item>().ToList();

}

public async Task<IEnumerable<Item>> GetItemsAsync(bool forceRefresh = false)

{

return await Task.FromResult(items);

}

}

}

## ViewModels\AboutViewModel.cs

## ViewModels\BaseViewModel.cs

## ViewModels\ItemDetailViewModel.cs

## ViewModels\ItemsViewModel.cs

## Views\AboutPage.xaml.cs

## Views\Favorite.xaml.cs

## Views\ItemDetailPage.xaml.cs

## Views\ItemsPage.xaml.cs

## Views\MainPage.xaml.cs

## App.xaml.cs

## Views\AboutPage.xaml

## Views\Favorite.xaml

## Views\ItemDetailPage.xaml

## Views\ItemsPage.xaml

## Views\MainPage.xaml

## App.xaml

# ЛИСТ РЕГИСТРАЦИИ ИЗМЕНЕНИЙ

|  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| Лист регистрации изменений | | | | | | | | | |
| Номера листов (страниц) | | | | | Всего листов (страниц в докум.) | № документа | Входящий № сопроводительного докум. и дата | Подп. | Дата |
| Изм. | Измененных | Замененных | Новых | Аннулированных |
|  |  |  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |  |  |