Содержание

Введение 7

1 Характеристика места практики 9

2 Сведения об имеющейся вычислительной технике и программном обеспечении 10

3 Сведения о прохождении практики в качестве дублера инженерно-технического работника 12

4 Индивидуальное задание 13

4.1 Постановка задачи 13

4.2 Проектирование структуры программы 13

4.2.1 Desktop версия приложения 13

4.2.2 Мобильная версия приложения 14

4.3 Функциональное взаимодействие модулей программы 14

4.3.1 Desktop версия приложения 14

4.3.2 Мобильная версия приложения 16

4.4 Разработка unit-тестов 17

4.5 Работа с программой 18

4.5.1 Desktop версия приложения 18

4.5.2 Мобильная версия приложения 20

Заключение 22

Список использованных источников 23

Приложения

А – Основные процедуры desktop приложения 24

Б – Основные процедуры мобильного приложения 25

В – Код UnitTest 25

ВВЕДЕНИЕ

Целью производственной практики по ПМ.01 Разработка модулей программного обеспечения для компьютерных систем является закрепление и совершенствование приобретенных в процессе обучения профессиональных умений обучающихся по специальности, развитие общих и профессиональных компетенций, адаптация к конкретным условиям деятельности предприятий и организаций, освоение современных производственных процессов, обучение трудовым приемам, операциям и способам выполнения трудовых процессов.

В процессе прохождения производственной практики необходимо будет решить следующие задачи:

* приобрести практический опыт разработки алгоритма решения поставленной задачи и реализации его средствами автоматизированного проектирования, использования инструментальных средств на этапе отладки программного продукта и проведения тестирования программного модуля по определенному сценарию;
* приобрести практический опыт анализа алгоритмов, в том числе с применением инструментальных средств, разработки кода программного продукта на основе готовой спецификации на уровне модуля, осуществления рефакторинга и оптимизации программного кода;
* приобрести практический опыт разработки мобильных приложений;
* сформировать профессиональные умения и навыки;
* приобрести практические навыки в будущей профессиональной деятельности при выполнении функций дублера инженерно-технических работников;
* выполнить индивидуальное задание.

В результате прохождения производственной практики необходимо будет подтвердить уровень овладения видом деятельности «Разработка модулей программного обеспечения для компьютерных систем» и следующими профессиональными компетенциями:

ПК 1.1 Формировать алгоритмы разработки программных модулей в соответствии с техническим заданием.

ПК 1.2 Разрабатывать программные модули в соответствии с техническим заданием.

ПК 1.3 Выполнять отладку программных модулей с использованием специализированных программных средств.

ПК 1.4 Выполнять тестирование программных модулей.

ПК 1.5 Осуществлять рефакторинг и оптимизацию программного кода.

ПК 1.6 Разрабатывать модули программного обеспечения для мобильных платформ.

Производственная практика проводится на базе Управления образования администрации муниципального образования «Теучежский район».

1 ХАРАКТЕРИСТИКА МЕСТА ПРАКТИКИ

Система образования муниципального образования «Теучежский район» оказывает значительное влияние на социально-экономическое развитие муниципалитета. Являясь самым большим потребителем бюджетных средств (более 50% бюджета района), система образования обеспечивает доступное бесплатное дошкольное, общее и дополнительное образование. В образовательных учреждениях имеют рабочие места 840 жителей Теучежского района. Система общего образования обеспечивает необходимый уровень подготовки выпускников школ для продолжения образования. Значительное место в работе образовательных учреждений занимает воспитательная работа, обеспечивающая досуг и развитие детей в вопросах краеведения, экологии, права, здорового образа жизни, патриотизма, уважения к старшим поколениям. Обеспечение отдыха и занятости детей в дни каникул способствует укреплению здоровья детей и сдерживает подростковую преступность и правонарушения. Наличие информационного центра в базовой школе (СОШ№1), Интернета во всех школах, детских садах и учреждениях дополнительного образования детей обеспечивает создание единого информационного пространства. Директор организации– Блягоз Н.Ш. кандидат педагогических наук, Почетный работник среднего профессионального образования Российской Федерации.

Юридический адрес: Октябрьская ул., 33, аул Понежукай

Руководитель организации – Блягоз Н.Ш.

2 СВЕДЕНИЯ ОБ ИМЕЮЩЕЙСЯ ВЫЧИСЛИТЕЛЬНОЙ ТЕХНИКЕ И ПРОГРАММНОМ ОБЕСПЕЧЕНИИ

Практика проходила в кабинете 4.

Кабинет 4 имеет следующие характеристики:

* площадь – 54,3 кв.м;
* электрощит – 1 шт.;
* электророзетки – 3 шт.;
* рабочие столы:
  + компьютерные – 2 шт.;

В кабинете установлены 2 компьютера со следующими характеристиками:

* материнская плата: ASUS;
* процессор: 3.21 ГГЦ;
* видео карта: AMD Athlon X2 260;
* оперативная память (RAM): 8 ГБ;
* жесткий диск (HDD): 500 ГБ;

Компьютеры объединены в локальную сеть и имеют выход в Internet.

Программное обеспечение рабочего места:

* Microsoft Visio Professional 2019;
* Microsoft Office 2019;
* Git 2.32;
* .NET Framework Developer Pack 4.8;
* Microsoft Visual Studio 2022 Community, включая следующие компоненты:
  + .NET desktop development Workload;
  + Universal Windows Platform development Workload;
  + Data storage and processing Workload;
  + Entity Framework (EF); IIS, ASP.NET and web development, Mobile development with .NET;
* Android SDK 30.29;
* SQL Server Management Studio 18.9.1;
* Visual Studio Code 1.57;
* Notepad++ 8.1;
* Postman 8.6.2.

Программное обеспечение сервера:

* Microsoft Visio Professional 2019;
* Microsoft Office 2019;
* Git 2.32;
* .NET Framework Developer Pack 4.8;
* Microsoft Visual Studio 2022 Community, включая следующие компоненты:
  + .NET desktop development Workload;
  + Universal Windows Platform development Workload;
  + Data storage and processing Workload;
  + Entity Framework (EF); IIS, ASP.NET and web development, Mobile development with .NET;
* SQL Server Management Studio 18.9.1;
* Visual Studio Code 1.57;
* Notepad++ 8.1;
* Postman 8.6.2;
* Сервер Gogs;
* IIS Express 10.0, Nginx 1.21, Apache 2.4.48;
* Microsoft SQL Server 2017 Express;

В 4 кабинете проводятся настройки и управление данными о сотрудниках и реквизитах организации.

3 СВЕДЕНИЯ О ПРОХОЖДЕНИИ ПРАКТИКИ В КАЧЕСТВЕ ДУБЛЕРА ИНЖЕНЕРНО-ТЕХНИЧЕСКОГО РАБОТНИКА

Во время прохождения практики в качестве дублера техника-программиста я выполнял следующие действия:

* ознакомление с системным и прикладным программным обеспечением на предприятии;
* работа с документацией;
* сбор и структурирование материала по теме индивидуального задания «Информационная система «Самбо»»;
* анализ алгоритмов, в том числе с применением инструментальных средств;
* разработка кода программного продукта на основе готовой спецификации на уровне модуля;
* написание программного кода, реализующего функциональность разрабатываемого приложения;
* разработка интерфейса приложения;
* отладка программного продукта с использованием спе­циализированных программных средств;
* разработка тестовых наборов и тестовых сценариев;
* осуществление рефакторинга и оптимизации программного кода;
* тестирование программного продукта;
* разработка технической документации к программному продукту;
* обслуживание оргтехники;
* установка и настройка необходимого программного обеспечения;
* разработка кода по заданию руководителя практики.

Выполненная работа соответствует профессиональным компетенциям, предусмотренным программой производственной практики ПМ.01 специальности 09.02.07 Информационные системы и программирование.

4 ИНДИВИДУАЛЬНОЕ ЗАДАНИЕ

4.1 Постановка задачи

Необходимо разработать приложение для ИС «Самбо».

Программа должна иметь графический интерфейс, быть проста в использовании и обеспечивать следующие возможности:

* хранения в файлах информацию о всех правилах и истории такого вида спорта, как самбо;
* интерфейс должен быть последователен и понятен;
* просмотр информации предоставленной в файлах программы.

Входными данными будут:

* выбранный тип правил;
* элемент меню для открытия нужной информации.

Выходными данными будут:

* страница с нужной для пользователя информацией;
* данные выбранного типа правил.

Также необходимо разработать пакет unit-тестов для функции, вычисляющей значения исследуемой функции.

4.2 Проектирование структуры программы

4.2.1 Desktop версия приложения

Разработанное приложение имеет следующие окна:

* главное окно, которое содержит области:
* боковое меню, которое позволит быстро перемещаться по всем страницам программы;
* список с правилами;
* страница с историей спорта;
* сведения о программе.

Обращение к окнам реализовано с помощью меню.

4.2.2 Мобильная версия приложения

Разработанное приложение имеет следующие окна:

* окно заставки;
* главное окно:
* окно с информацией о спорте;
* окно с правилами;

Обращение к окнам реализовано с помощью меню (за исключением окна заставки).

4.3 Функциональное взаимодействие модулей программы

4.3.1 Desktop версия приложения

На основе разработанного технического проекта был написан программный продукт «Информационная система «Самбо»». Для создания приложения была использована среда программирования Visual Studio 2022 и язык программирования С#. В программе используются следующие языковые модули:

* System.Windows.Controls− предоставляет классы для создания элементов, известных как элементы управления, позволяющих пользователю взаимодействовать с приложением;
* System.Diagnostics− содержит типы, позволяющие взаимодействовать с системными процессами, журналами событий и счётчиками производительности;
* System− содержит фундаментальные и базовые классы, определяющие часто используемые типы значений и ссылочных данных, события и обработчики событий, интерфейсы, атрибуты и исключения обработки;
* System.Windows− содержит типы, используемые в приложениях WindowsPresentationFoundation (WPF), включая клиенты анимации, элементы управления пользовательского интерфейса, привязку данных и преобразование типов;
* CefSharp− содержит типы, поддерживающие компиляцию и создание кода на языке C#, а также поддерживающие взаимодействие между средой DLR и языком C#;
* System.IO −cодержит типы, поддерживающие ввод и вывод, включая возможности чтения и записи данных в потоках как синхронно, так и асинхронно, реализации пользовательского ведения журнала и обработки входящих и исходящих потоков данных в последовательных портах;
* System.Data− содержит типы для доступа к данным из различных источников и для управления этими данными. Пространство имен верхнего уровня и несколько дочерних пространств имен образуют архитектуру ADO.NET и поставщиков данных ADO.NET [9].
* System.Threading− содержат типы, обеспечивающие возможности многопотокового программирования. Дочернее пространство имен предоставляет типы, которые упрощают задачу написания параллельного и асинхронного кода;
* System.Collections.Generic – работа с обобщёнными коллекциями (List<string>) [5].

В программе были созданы следующие пользовательские модули:

* AboutProgramPageViewModel.cs (страница с информацией о программе) – информация о программе;
* MainWindowViewModel.cs (главное окно программы) − обеспечивает возможность перемещения между страницами программы;
* HistoryPageViewModel.cs (страница с историей спорта) – информация о истории спорта;
* RulesPageViewModel.cs (страница с правилами спорта) – модуль который позволяет просматривать правила;
* FrameManager.cs - класс для управления страницами в приложении.

В модуле FileManager.cs описаны и используются следующие процедуры:

* public async Task<string> ReadFile(string name) − производит поиск и считывание нужной информации из файла;
* public Uri FindUri(string name) − ищет нужную картинку и отправляет ее путь;

Код данных процедур представлен в приложении А.

4.3.2 Мобильная версия приложения

На основе разработанного технического проекта была написана мобильная версия программного продукта «Информационная система «Самбо»». Для создания приложения была использована среда программирования Android Studio и язык программирования Kotlin. В программе используются следующие языковые модули:

* android.content.Context - это базовый абстрактный класс, реализация которого обеспечивается системой Android. Этот класс имеет методы для доступа к специфичным для конкретного приложения ресурсам и классам и служит для выполнения операций на уровне приложения, таких, как запуск активностей, отправка широковещательных сообщений, получение намерений и прочее;
* android.view.View - компонент(виджет), который рисуется на экране;
* java.util.ArrayList - класс, позволяющий создавать списки;
* android.app.AlertDialog - это уведомление, которое появляется, когда определенное приложение использует систему Android для размещения важной информации для пользователя;
* android.content.Intent - класс, объекты которого используются для взаимодейтвия между отдельными частями Android приложения, либо между различными приложениями системы;
* android.os.Bundle - необходим для временного хранения данных в процессе выполнения;
* android.view.Menu – класс, представляющий меню в приложениях;
* android.view.MenuItem – класс, отвечающий за пункты меню;
* android.widget.Toast - это небольшое, всплывающее сообщение, появляющееся внизу и экрана на непродолжительное время;
* androidx.appcompat.app.AppCompatActivity – это базовый класс для действий, которые используют функциональные возможности панели поддержки библиотеки;
* android.widget.EditText - это стандартный виджет ввода текста в приложениях Android;
* java.io.FileOutputStream - поток вывода, который содержит методы, записывающие данные в файл;
* android.os.AsyncTask - предлагает простой и удобный механизм для перемещения трудоёмких операций в фоновый поток.

В программе были созданы следующие пользовательские модули:

* ActivityPreview.kt – окно заставки;
* MainActivity.kt – основное окно, в котором находится информация о виде спорта «Самбо»;
* RulesActivity.kt – окно для ввода правил;

В модуле RulesActivity.kt описаны и используются следующие процедуры:

* fun readFile(args: Array<String>): String – метод для чтения файла.

В модуле MainActivity.kt описаны и используются следующие процедуры:

* fun onClick(View : view) – открывает окно с правилами;

Код класса RulesActivity.kt представлен в приложении Б.

4.4 Разработка unit-тестов

Для осуществления корректной работы разработанного приложения, в части правильности вычисления значений функции, был разработан тестовый проект, включающий в себя набор unit-тестов. Данные тесты охватывают все возможные тестовые случаи, которые необходимо проверить:

* чтение файла c корректными данными;
* чтение файла c неправильными данными;
* поиск картинки по неправильному имени;

Код данных тестов представлен в приложении В.

4.5Работа с приложением

4.5.1 Desktop версия приложения

При запуске программы открывается главное окно (рисунок 1).



Рисунок 1 – Главное окно

Интерфейс главного окна имеет следующие элементы:

* боковое меню, которое содержит три кнопки («О программе», «История», «Правила»);
* блок с информацией и отображения страницы;
* заголовок окна.

При открытии окна автоматически открывается страница с описанием программы.

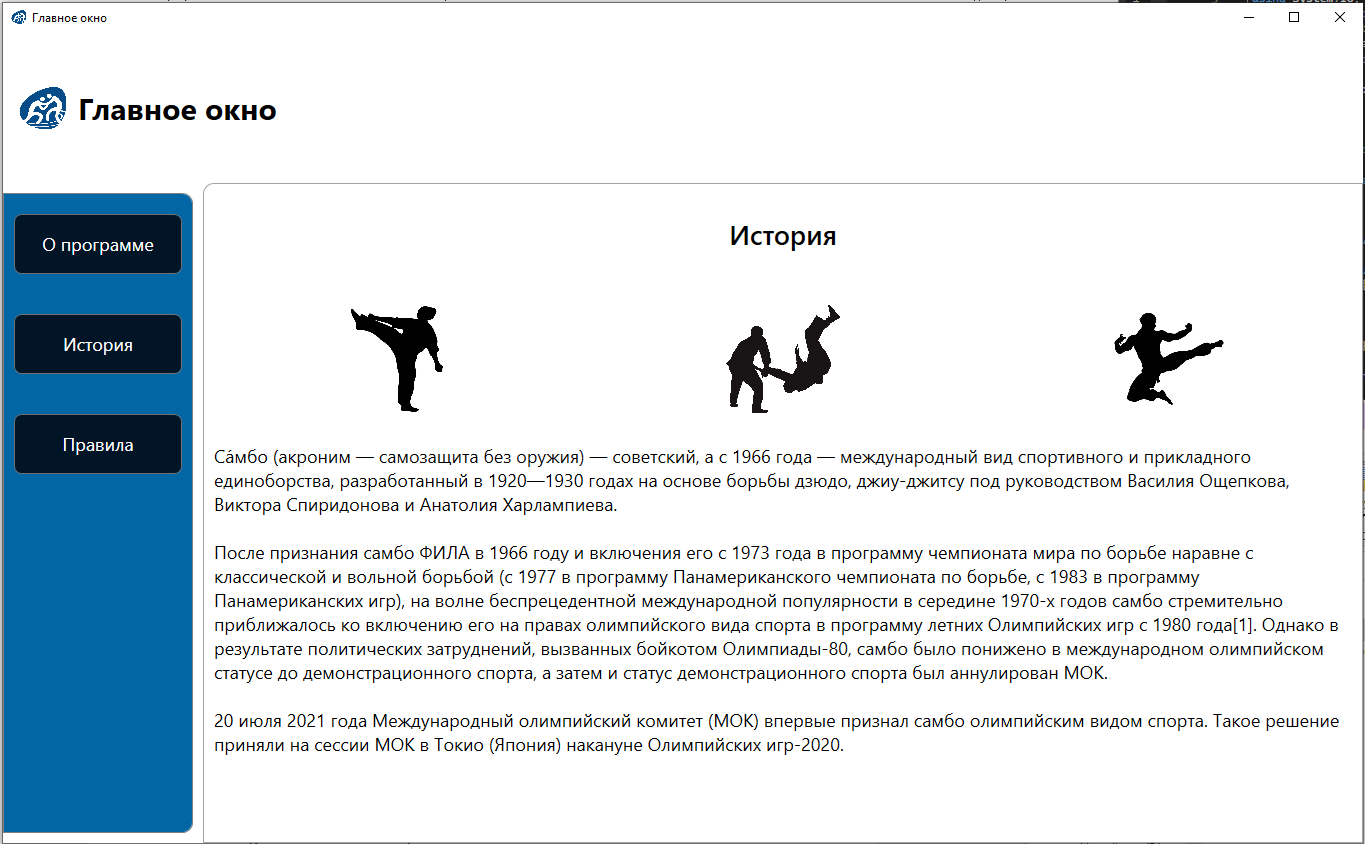


Рисунок 2 – Страница «История»

При нажатии на кнопку «История», откроется окно с информацией о данном виде спорта.

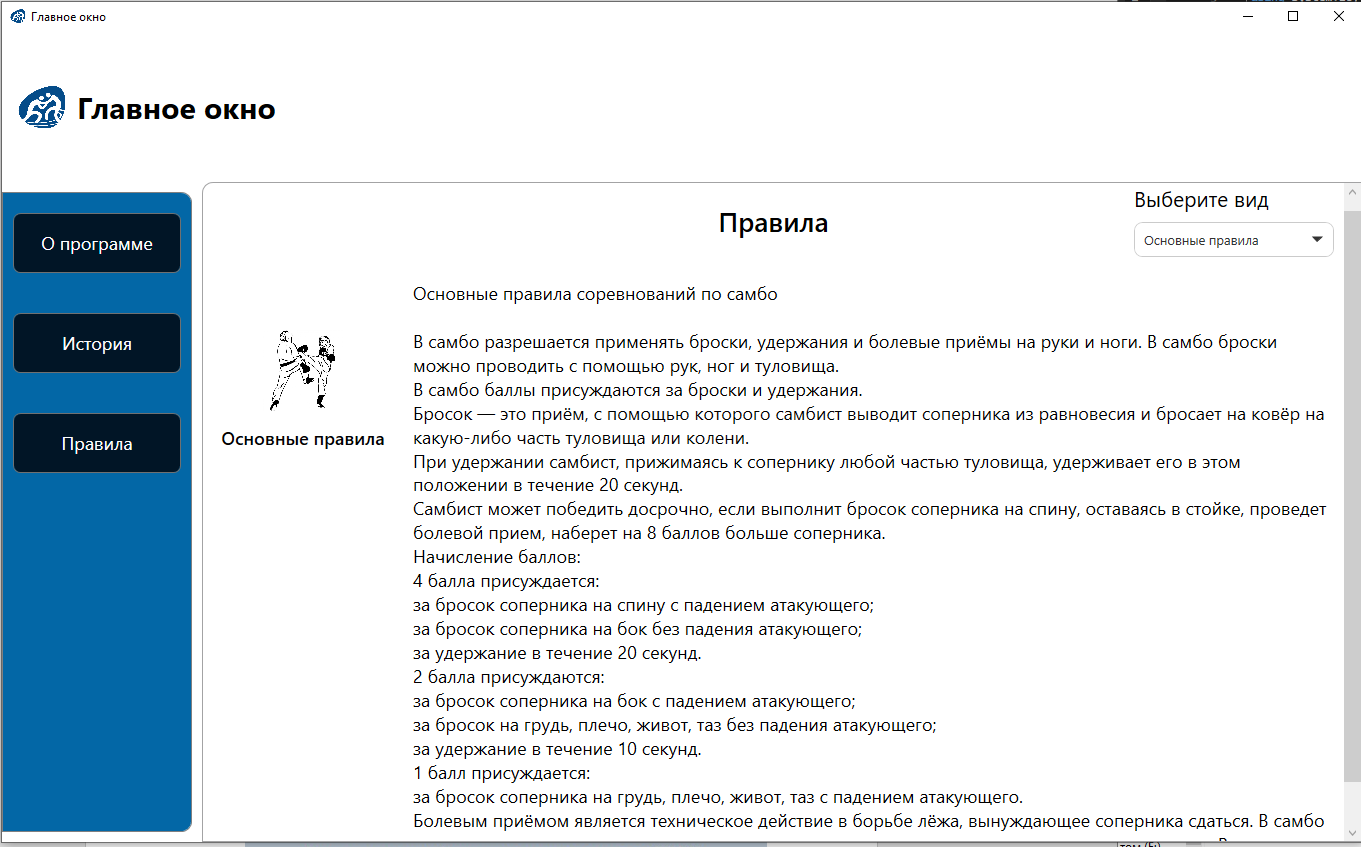


Рисунок 3 – Страница «Правила»

При нажатии пункта меню «Правила» в главном окне, откроется страница с различными правилами. Для просмотра определенных правил вам необходимо выбрать интересующий вас пункт из выпадающего списка.

4.5.2 Мобильная версия приложения

При запуске программы открывается заставка (рисунок 4).

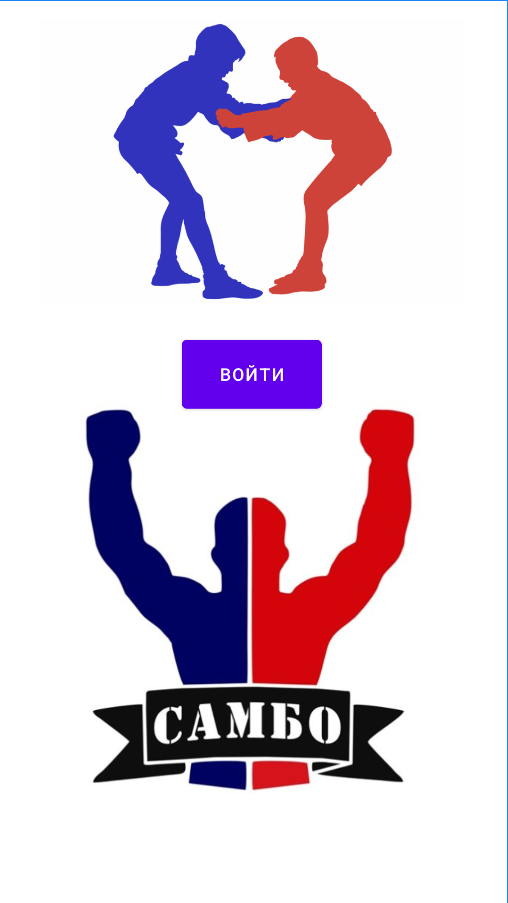


Рисунок 4 – Заставка

После того как вы нажмете на кнопку «Войти», откроется главное окно приложения (рисунок 5).



Рисунок 5 – Основное окно программы

На данном окне вы можете ознакомиться с главными терминами и основной информацией о данном виде спорта. Далее вы можете нажать на кнопку «Правила» (рисунок 6).

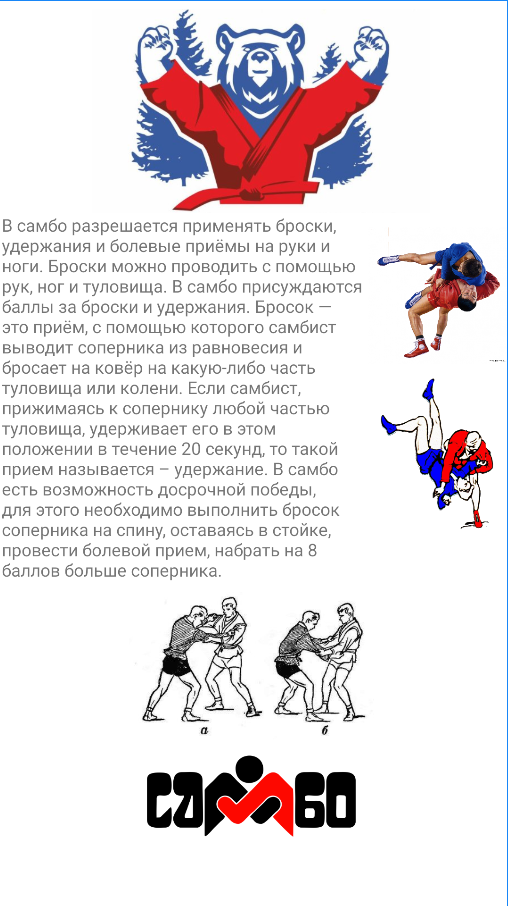


Рисунок 6 – Окно «Правила»

В данном окне вы уже можете ознакомиться с правилами Самбо и насладиться прекрасным интерфейсом.

ЗАКЛЮЧЕНИЕ

В результате прохождения производственной практики мною были закреплены теоретические знания, полученные при изучении ПМ.01; отработаны профессиональные компетенции по получаемой специальности; приобретены практические навыки в будущей профессиональной деятельности; выполнено индивидуальное задание.

Мною было разработано приложение создания ИС «Сумо».

Программа имеет удобный и понятный пользователю графический интерфейс, проста в использовании и обеспечивает следующие возможности:

* хранение в файлах информации о всех правилах и истории такого вида спорта, как самбо;
* интерфейс последователен и понятен;
* просмотр информации предоставлен в файлах программы.

В ходе выполнения разработки данного приложения были:

* проанализирована тема «Самбо»;
* спроектирована структура приложения и программного кода;
* разработаны справочные материалы;
* выбраны средства реализации приложения;
* разработан программный код приложения;
* разработан тестовый проект, реализующий оптимальный набор unit-тестов.

Основными достоинствами программы являются:

* программа проста в установке, настройке и эксплуатации;
* данный программный продукт предназначен для широкого круга пользователей;
* наличие справочной системы.

К недостаткам данной программы можно отнести то, что она работает с файлами. В дальнейшем планируется перевести ее на работу с базой данных.

СПИСОК ИСПОЛЬЗОВАННЫХ ИСТОЧНИКОВ

1. Албахари Б., Албахари Дж. С# 7.0. Справочник. Полное описание языка. М.: Вильямс, 2018. 1024 с.
2. Виссер Д. Разработка обслуживаемых программ на языке С#. М.: ДМК Пресс, 2017. 194с.
3. Джепикс Ф., Троелсен Э. Язык программирования C# 7 и платформы .NET и .NET Core. М.: Вильямс, 2018. 1328 с.
4. Мюллер Д. П., Семпф Б., Сфер Ч. С# Для чайников. М.: Диалектика, 2016. 608 с.
5. Скит Д. С# Для профессионалов. Тонкости программирования. М.: Вильямс, 2019. 608 с.
6. Документация Microsoft. URL: https://docs.microsoft.com/ru-ru/documentation (дата обращения: 01.12.2021).
7. Библиотека классов .NET Framework. URL: https:// msdn.microsoft.com/ru-ru/library/mt472912(v=vs.110).aspx (дата обращения: 01.12.2021).
8. WPF Globalization and Localization Overview: https://learn.microsoft.com/en-us/dotnet/desktop/wpf/advanced/wpf-globalization-and-localization-overview?view=netframeworkdesktop-4.8
9. Общие сведения о платформе Android. URL: https://developer.android.com/guide (дата обращения: 30.11.2021).

ПРИЛОЖЕНИЕ А

Основные процедуры desktop приложения

public async Task<string> ReadFile(string name)

{

try

{

var path = \_path + "Files/" + name + ".txt";

using (StreamReader reader = new StreamReader(path))

{

var result = await reader.ReadToEndAsync();

return result;

}

}

catch

{

return " ";

}

}

public Uri FindUri(string name)

{

try

{

DirectoryInfo directoryInfo= new DirectoryInfo(\_path + "Images/");

foreach (var item in directoryInfo.GetFiles())

{

if(item.Name == name + ".png")

{

return new Uri(item.FullName);

}

}

return null;

}

catch

{

throw;

}

}

ПРИЛОЖЕНИЕ B

Основные процедуры мобильного приложения

fun readFile(args: Array<String>) : String {

val bufferedReader: BufferedReader = File("example.txt").bufferedReader()

val inputString = bufferedReader.use { it.readText() }

return inputString

}

ПРИЛОЖЕНИЕ C

Код UnitTest

[TestClass]

public class UnitTest

{

[TestMethod]

public void ReadFile\_NoramlName\_Ok()

{

//Arange

FileManager fileManager = new FileManager();

string name = "AboutProgram";

var expected = "Эта программа предназначена для экскурса в такой вид спорта, как самбо. \r\n\r\nЗдесь вы можете ознакомиться с историей и правилами этого пркрасного элемента борьбы."

//Act

var result = fileManager.ReadFile(name);

//Assert

Assert.AreEqual(expected, result);

}

[TestMethod]

public void ReadFile\_InvalidName\_SpaceString()

{

//Arange

FileManager fileManager = new FileManager();

string name = "Abouram";

var expected = " ";

//Act

var result = fileManager.ReadFile(name);

//Assert

Assert.AreEqual(expected, result);

}

[TestMethod]

public void FindUri\_InvalidName\_SpaceString()

{

//Arange

FileManager fileManager = new FileManager();

string name = "Основе правила";

var expected = " ";

//Act

var result = fileManager.FindUri(name);

//Assert

Assert.AreEqual(expected, result);

}

}