**Содержание**

[Введение……………………………………………...………………………….4](#_Toc75523873)

[1 Анализ задачи 6](#_Toc75523874)

[1.1 Постановка задачи 6](#_Toc75523875)

[1.2 Инструменты разработки 6](#_Toc75523876)

[1.3 Требования к приложению 7](#_Toc75523877)

[2 Проектирование задачи 9](#_Toc75523878)

[2.1 Организация данных 9](#_Toc75523879)

[2.2 Процессы 9](#_Toc75523880)

[2.3 Описание внешнего пользовательского интерфейса 9](#_Toc75523881)

[3 Реализация 1](#_Toc75523882)2

[3.1.1 Структура программы 1](#_Toc75523883)2

[3.1.2 Структура и описание процедур и функций пользователя 1](#_Toc75523884)2

[3.1.3 Описание использованных компонентов 1](#_Toc75523885)3

[3.2 Спецификация программы 1](#_Toc75523886)4

[4 Тестирование 1](#_Toc75523887)6

[5 Руководство пользователя 1](#_Toc75523888)8

[5.1 Общие сведения о программном продукте 1](#_Toc75523889)8

[5.2 Инсталляция 1](#_Toc75523890)8

[5.3 Выполнение программы](#_Toc75523891) 19

[5.3.1 Запуск программы](#_Toc75523892) 19

[5.3.2 Инструкции по работе с программой](#_Toc75523893) 19

[5.3.3 Завершение работы с программой 2](#_Toc75523894)3

[5.4 Использование системы справочной информации 2](#_Toc75523895)3

[Заключение 2](#_Toc75523896)4

[Список использованных источников 2](#_Toc75523897)5

[Приложение А 2](#_Toc75523898)6

[Листинг программы 2](#_Toc75523899)6

**Введение**

На курсовое проектирование была поставлена задача разработать познавательное приложение по теме «Музеи Гродно».

Цель курсового проекта заключается в знакомстве массового пользователя с музеями города Гродно.

Создаваемое познавательное приложение рассчитано на любого рода пользователей. Применить его смогут не только люди, проживающие в Гродно, но и те, кто готовится к путешествию и планирует узнать город Гродно.

Далее приведем краткое описание разделов пояснительной записки.

Первый раздел носит название “Анализ задачи”. В нем вы сможете ознакомиться с постановкой задачи, которая включает в себя: исследование предметной области поставленной задачи, определение ее организационно-экономической сущности. Также в этом разделе вы сможете узнать о том, как данная задача решается в настоящее время. Все входные и выходные данные тоже будут описаны в первом разделе. В подразделе “Инструменты разработки” будет рассмотрена среда, в которой создается данный курсовой проект. Здесь также будут установлены минимальные и оптимальные требования к аппаратным характеристикам, обеспечивающим правильное функционирование поставленной задачей.

В разделе “Проектирование задачи” будут рассмотрены основные аспекты разработки программного продукта. Здесь можно будет узнать об организации данных в контексте среды разработки. В данном разделе будет четко описан пользовательский интерфейс, составлены алгоритмы процесса обработки информации, описана разработка системы справочной информации.

“Реализация задачи” – это третий раздел пояснительной записки, в котором описываются все элементы и объекты, которые будут использованы при реализации данного приложения. В этом разделе будут четко описаны функции пользователя и их структура. Здесь можно будет найти таблицу, в которой будет представлена полная аннотация файлов используемых в данном проекте.

Четвертый раздел – “Тестирование”. В нем будет описано полное и функциональное тестирование данной программы, т.е. будет оттестирован каждый пункт меню, каждая операция, которая выполняется приложением. Будут смоделированы все возможные действия пользователя при работе с программой, начиная от запуска до выхода.

В разделе “Применение” будет описано назначение, область применения, среда функционирования курсовой программы. Также в нем будет описано использование справочной системы.

“Заключение” будет содержать краткую формулировку задачи, результаты проделанной работы, описание использованных методов и средств, описание степени автоматизации процессов на различных этапах разработки.

В “Литературе” будет приведен список используемых при разработке источников.

В приложениях к пояснительной записке будет приведен листинг программы с необходимыми комментариями.

Схема работы системы будет представлена в графической части.

## **1 Анализ задачи**

## **1.1 Постановка задачи**

**Наименование задачи:** Разработка познавательного приложение «Музеи Гродно». Создаваемое приложение будет рассчитано для пользователей любого возраста и рода занятия.

**Цель разработки:** Знакомство массового пользователя с музеями города Гродно.

**Назначение:** Создаваемое познавательное приложение рассчитано на любого рода пользователей. Применить его смогут не только люди, проживающие в Гродно, но и те, кто готовится к путешествию и планирует узнать город Гродно.

**Периодичность использования:** зависит от нужд потребителя, но скорее всего будет использовано один раз.

**Источники и способы получения данных:** теоретические материалы про музеи Гродно.

Разрабатываемый программный продукт должен позволять выполнить следующие действия:

- познание информации по музеям;

- просмотр расположения на карте;

- проверка знаний пользователя.

## **1.2 Инструменты разработки**

**Delphi 10.4** – главная среда разработки программы, в которой прописывается код и создаются формы.

**Microsoft Word 2010** – редактор текста для написания документации.

**Microsoft Power Point 2010** – программа для создания презентации.

**Adobe Photoshop CS 6** – графический редактор.

**Браузер Google Chrome** – для нахождения информации о создании игровых приложений в среде Delphi 10.4, поиск информации о создании интеллектуальных приложеня, медиафайлов.

**Блокнот** – программа была использована для написания ini-файлов.

**Brackets –** программа для написания HTML кода.

**Smart Install Maker –** программа для создания установщика приложения.

**Dr.Explain –** программа для создания справочной системы.

Для оптимального функционирования разрабатываемого приложения необходимо наличие у компьютера следующих параметров:

− 1.5 GHz процессор (или более быстрый);

− 1 GB RAM;

− от 400 MB доступного места на жестком диске.

Разработка ведется на ноутбуке Lenovo. У данного ноутбука следующие параметры:

-процессор AMD Ryzen 5 4600H 3.0GHz;;

-объем ОЗУ 8 гб;

- объем места на HDD – 512 мб ;

-ОС – Windows 10.

Как видно разрабатываемое приложение не очень требовательно к аппаратным ресурсам, что, является большим плюсом.

## **1.3 Требования к приложению**

На этапе исследования предметной области был установленный целый ряд требований, которые предъявляют к разрабатываемой задаче.

При моделировании форм следует учесть такие моменты:

* интуитивно понятный интерфейс.
* небольшое количество информации на каждой форме.

Так же следует обратить внимание на такие моменты:

* большие окна форм;
* визуально понятный и приятный глазу интерфейс;
* средние кнопки.

Минимальные системные требования:

− 1.5 GHz процессор (или более быстрый);

− 1 GB RAM;

− от 400 MB доступного места на жестком диске.

Рекомендуемые:

− двухъядерный AMD Athlon или аналогичный процессор Intel (или более быстрый) ;

− 2 GB RAM ;

− от 400 MB доступного места на жестком диске.

**Требования к надежности:**

Специальных требований к надежности не предъявляется.

Время восстановления после отказа, вызванного сбоем электропитания технических средств (иными внешними факторами), не фатальным сбое (не крахом) операционной системы, не должно превышать 30-ти минут при условии соблюдения условий эксплуатации технических и программных средств.

## 

## **2 Проектирование задачи**

## **2.1 Организация данных**

Основными средствами хранения информации в приложении, будут типизированные и текстовые файлы, в них будет храниться теоретическая информация и параметры настроек.

Система справочной информации будет представлена файлом справки, который будет содержать информацию о приложении и правила его использования.

Основными функциями приложения будут являться:

− изучение теоретического материала;

− просмотр карты расположения;

− просмотр фото музеев;

− прохождение теста.

## **2.2 Процессы**

Согласно всем перечисленным требованиям и указаниям, которые были рассмотрены в разделе «Анализ задачи», было определено, чем конкретно должна заниматься разрабатываемая программа. Главной её задачей будет являться знакомство пользователя с музеями города Гродно.

Для реализации задач будут использоваться процедуры.

## **2.3 Описание внешнего пользовательского интерфейса**

Важным при выполнении курсового проекта является организация диалога между пользователем и самой программой. Во многом это зависит от того, как программист разработает данную программу, какие компоненты будут использованы и какие методы будут автоматизированы.

Особое внимание следует уделить интерфейсу. Разработчик должен так организовать внешний вид своей программы, что бы пользователь понял, что от него требуется.

Для организации эффектной работы пользователя нужно создать целостное приложение данной предметной области, в которой все компоненты приложения будут сгруппированы по функциональному назначению. При этом необходимо обеспечить удобный графический интерфейс пользователя.

Я постарался так и сделать. Весь интерфейс приятен, не режет глаза, и прост в использовании.

Таким образом, для успешной работы всего проекта в целом следует обеспечить интуитивно понятный интерфейс с приятной гаммой цветов и шрифтами. Примерный интерфейс приложения можно увидеть на рисунке 1.

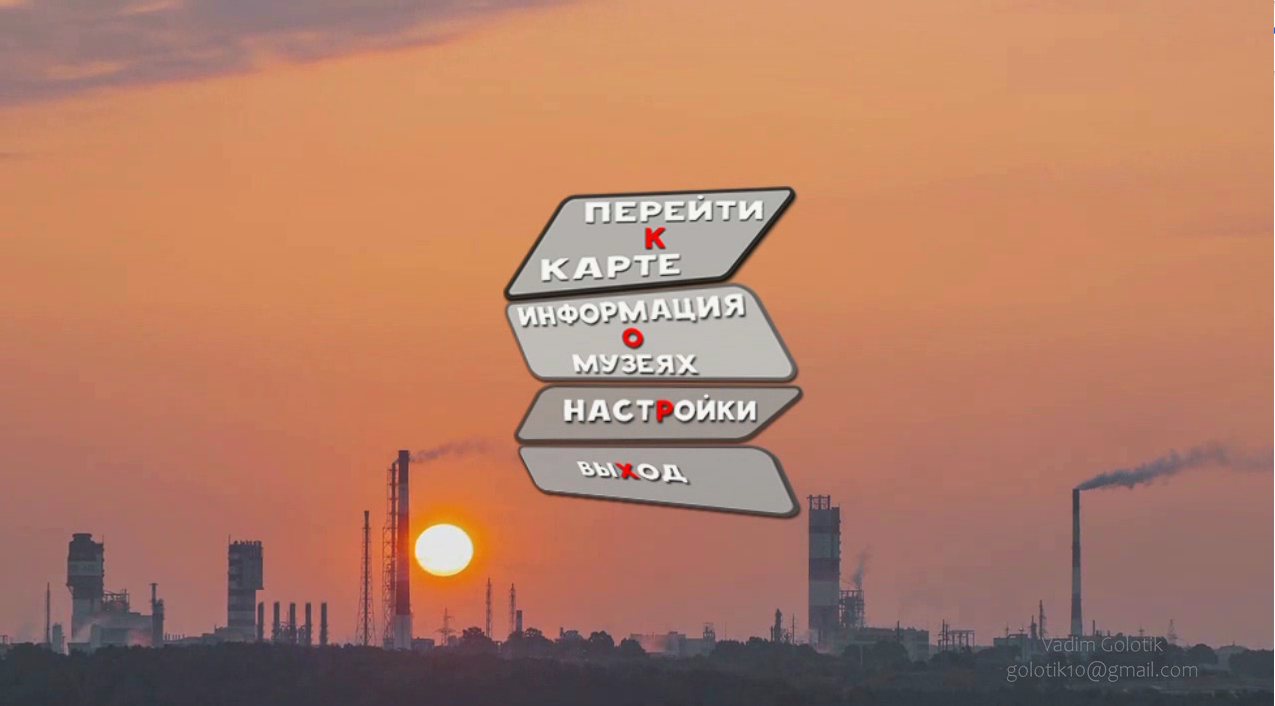


Рисунок 1 – Примерный вид формы

Структура навигации по проекту представлена на рисунке 2:



Рисунок 2 – Структура навигации по проекту

## 

## 

## 

## **3 Реализация**

## **3.1.1 Структура программы**

Данный курсовой проект содержит 5 модулей. Далее рассмотрим назначение каждого модуля:

Модуль Unit2 является модулем, за которым закреплена главная форма. С этой формы можно перейти на модули Unit3, Unit4, Unit 5, Unit 6.

Модуль Unit1 – загрузочный экран, который запускается при запуске приложения.

Unit3 - является модулем, который содержит графическое изображение карты города, в свою очередь на которой расположены музеи.

Unit4 - модуль, в котором находятся настройки приложения.

Unit5 - модуль, который позволяет пользователю просмотреть информация сразу обо всех музеях.

Unit6 - модуль, в котором расположен тест.

## **3.1.2 Структура и описание процедур и функций пользователя**

Описание разработанных процедур находятся в таблице 1.

Таблица 1 – Процедуры и функции

|  |  |  |  |
| --- | --- | --- | --- |
| Имя процедуры (функции) | В каком модуле находится | За каким компонентом закреплена | Назначение |
| 1 | 2 | 3 | 4 |
| procedure TForm1.FormCreate(Sender: TObject); | Form1 | FormCreate | Воспроизведение музыки |
| procedure TForm1.Timer1Timer(Sender: TObject); | Form1 | Timer1Timer | Таймер для полосы загрузки |
| procedure TForm2.FormActivate(Sender: TObject); | Form2 | FormActivate | При активации формы вопроизводить видео |
| procedure TForm2.Image1Click(Sender: TObject); | Form2 | Image1Click | Переход на другую форму |
| Продолжение таблицы 1 | | | |
| 1 | 2 | 3 | 4 |
| procedure TForm2.Image2Click(Sender: TObject); | Form2 | Image2Click | Переход на другую форму |
| procedure TForm2.Image3Click(Sender: TObject); | Form2 | Image3Click | Переход на другую форму |
| procedure TForm2.Image4Click(Sender: TObject); | Form2 | Button4Click | Переход на другую форму |
| procedure TForm2.N1Click(Sender: TObject); | Form2 | N1Click | Вызов справочного меню |
| procedure TForm2.N2Click(Sender: TObject); | Form2 | N2Click | Кнопка выхода |
| procedure TForm2.N3Click(Sender: TObject); | Form2 | N3Click | Переход на форму с тестом |
| procedure TForm3.Image1-9Click(Sender: TObject); | Form3 | Image1-9Click | Переход на другую форму |
| procedure TForm4.Image1Click(Sender: TObject); | Form4 | Image1Click | Сохранение настроек |
| procedure TForm5.Image1-10Click(Sender: TObject); | Form5 | Button1-10Click | Смена информации и фото музеев |
| procedure TForm6.Image1Click(Sender: TObject); | Form6 | Image1Click | Принятие ответа |
| procedure TForm6.Image3Click(Sender: TObject); | Form6 | Image3Click | Вывод вопросов на экран |
| procedure TForm6.RadioGroup1Click(Sender: TObject); | Form6 | RadioGroup1Click | Выбор варианта в тесте |

## **3.1.3 Описание использованных компонентов**

Описание использованных для разработки приложения компонентов приводится в таблице 2.

Таблица 2 – Использованные компоненты

|  |  |  |
| --- | --- | --- |
| Компонент | На какой форме расположен | Назначение |
| TCheckBox | Form4 | Используется для изменения настроек приложения |
| TTimer | Form1 | Используется для определения длительности загрузочного экрана |
| TWebBrowser | Form5 | Используется для отображения html-страниц с информацией о том или ином музее |
| TLabel | Form1 | Отображение текста на загрузочном экране |
| TImage | Все формы | Используется для более красивого и интересного интерфейса |
| TPanel | Form2 | Используется в виде дисплея для отображения видеофайлов |
| TGauge | Form1 | Отображение полосы загрузки на загрузочном экране |
| TMainMenu | Form2 | Используется для открытия справочной системы |
| TMediaPlayer | Form1,2 | Используется для воспроизведения аудио- и видеофайлов |
| TRadioGroup | Form 6 | Используется для вывода и ввода информации в тесте |
| TRadioButton | Form 6 | Используется для выбора варианта в тесте |

## **3.2 Спецификация программы**

Точное название проекта и его состав приводится в Таблице 3.

Таблица 3 – Спецификация программы

|  |  |
| --- | --- |
| Имя файла | Назначение |
| 1 | 2 |
| Project1.exe | Исполняемый файл проекта |
| Project1.dpr | Файл проекта, связывает все файлы, из которых состоит приложение |
| Unit1.pas | Файл программного модуля для формы Form1 |
| Unit2.pas | Файл программного модуля для формы Form2 |
| Unit3.pas | Файл программного модуля для формы Form3 |
| Продолжение таблицы 3 | |
| 1 | 2 |
| Unit4.pas | Файл программного модуля для формы Form4 |
| Unit5.pas | Файл программного модуля для формы Form5 |
| Setup.exe | Установочный файл |
| Help.chm | Файл справки, содержит помощь по работе с программой |
| Unit1.dfm | Форма загрузочного экрана |
| Unit2.dfm | Форма главного меню, с переходом на другие формы |
| Unit3.dfm | Форма с графическим изображением карты |
| Unit4.dfm | Форма с настройками |
| Unit5.dfm | Форма с информацией музеев |
| \*.html | Файлы с информацией о группах |
| \*.wav, \*wmv | Видеофайлы, аудиофайлы |
| \*jpg,\*.png,\*.jpeg | Изображения |
| \*.chm | Справочная система |
| Unit6.pas | Файл программного модуля для формы Form6 |
| Unit6.dfm | Форма с тестом |

**4 Тестирование**

При разработке данной программы многие возникающие ошибки и недоработки были исправлены на этапе реализации проекта. После завершения испытания реализации программы было проведено тщательное функциональное тестирование. Функциональное тестирование должно гарантировать работу всех элементов программы в автономном режиме.

Отчёт о результатах тестирования предоставлен в Таблице 4.

Таблица 4 – Отчёт результатах тестирования

|  |  |  |  |
| --- | --- | --- | --- |
| Тест | Ожидаемый результат | Физический результат | Результат тестирования |
| Проверка входа пользователя в систему | Открытие экрана | Открытие загрузочного экрана | Выполнено |
| Проверка отображения информации о музеях | В WebBrowser отображается выбранный материл | В WebBrowser отобразилась информация | Выполнено |
| Проверка кнопки «Тест» | Отображение теста | Отображение теста | Выполнено |
| Проверка кнопок «Информация о музеях» | Переход на форму с веб-браузером | Переход на форму с веб-браузером | Выполнено |
| Проверка пункта меню «Перейти к карте» | При нажатии на флажок осуществляется переход к музею | При нажатии на флажок осуществляется переход к музею | Выполнено |
| Проверка пункта меню «Настройки» | При нажатии на переключатель, изменяются настройки приложения | При нажатии на переключатель, изменяются настройки приложения | Выполнено |
| Проверка пункта меню «Выход» | При нажатии происходит выход из приложения | При нажатии происходит выход из приложения | Выполнено |
| Проверка пункта «Следующий вопрос» | При нажатии происходит вывод следующего вопроса теста | При нажатии происходит вывод следующего вопроса теста | Выполнено |
| Проверка пункта «Справка» | Открытие формы «Справка | Открытие формы «Справка | Выполнено |

Когда я делал свой программный продукт, я столкнулся с рядом проблем, которые исправил. Например: я не мог вставить видеоклип в заставку, но проблема решена, и пользователь теперь наслаждается красивой и качественной заставкой. Также была проблема с изменением настройки звуков нажатия на клавиши, но после изучения многих форумов, проблема была решена и программа полностью работает без сбоев.

Элементы программы были проверены, и было установлено, что все они работают правильно и выполняют задачи, указанные в процедурах.

При использовании данного познавательного приложения пользователь не столкнется с ошибками системы.

## **5 Руководство пользователя**

## **5.1 Общие сведения о программном продукте**

Разработанная программа является информативным приложением, с помощью которого пользователь сможет узнать больше информации об музеях Гродно.

Данный программный продукт предназначен для всех людей, независимо от возраста или пола. В программе они познакомятся историей, интересной информацией музеев в давних и нынешних годах.

Быстродействие любой программы во многом зависит от характеристик выбранного персонального компьютера: рабочей частоты процессора, объема оперативной памяти и т.д. Несмотря на все реализованные в ней задачи, она легко запускается и функционирует на любых машинах.

Тестирование проводилось на разных классах ЭВМ и работать с данной программой было комфортно. Программа разработана на ПК со следующими характеристиками:

* процессор Ryzen 5 4600H 3.0 Ghz ;
* объем ОЗУ 8Гб;
* графический адаптер NVIDIA Geforce GTX 1650 TI;
* операционная система Windows 10.

## **5.2 Инсталляция**

Для того, чтобы установить программу необходимо запустить файл Setup.exe. Появится окно установки электронной энциклопедии музыки.

При нажатии на кнопку «Далее» появляется окно выбора каталога установки.

Следуем инструкции и устанавливаем приложение.

## **5.3 Выполнение программы**

## **5.3.1 Запуск программы**

Данную программу можно запустить различными способами.

Первым из них является запуск с помощью ярлыка на рабочем столе. Необходимо дважды щелкнуть левой кнопкой мыши на ярлыке с названием Grodno Museum.

Вторым способом является запуск из каталога, в который устанавливалось приложение (по умолчанию C:\ProgramFiles(x86)\Berezich in comp.\Grodno Museum.

По подготовленным тестам будет осуществляться функциональное и полное тестирование программного продукта. Отчет о результатах тестирования будет представлен в 4 разделе пояснительной записки.

## **5.3.2 Инструкции по работе с программой**

После запуска на экране появляется загрузочный экран (Рисунок 3)

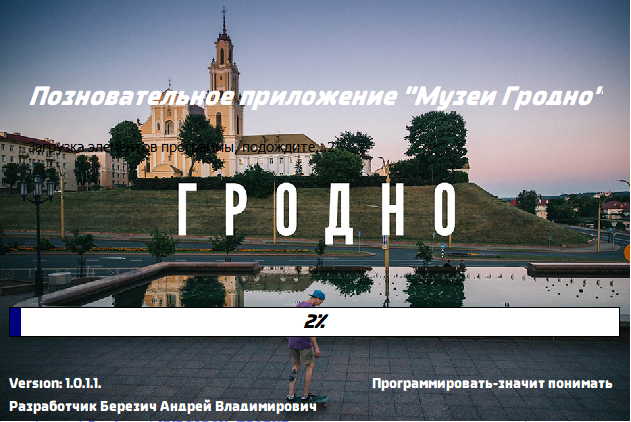


Рисунок 3 – Экран загрузки

После полной загрузки появляется главное меню (Рисунок 4)



Рисунок 4 – Главное меню

После нажатия на кнопку «Перейти к карте», пользователь перейдет на карту с музеями (Рисунок 5)



Рисунок 5 – Выбор «Перейти к карте»

При нажатии на кнопку «Информация о музеях»(Рисунок 6,7)



Рисунок 6 – Выбор «Информация о музеях»

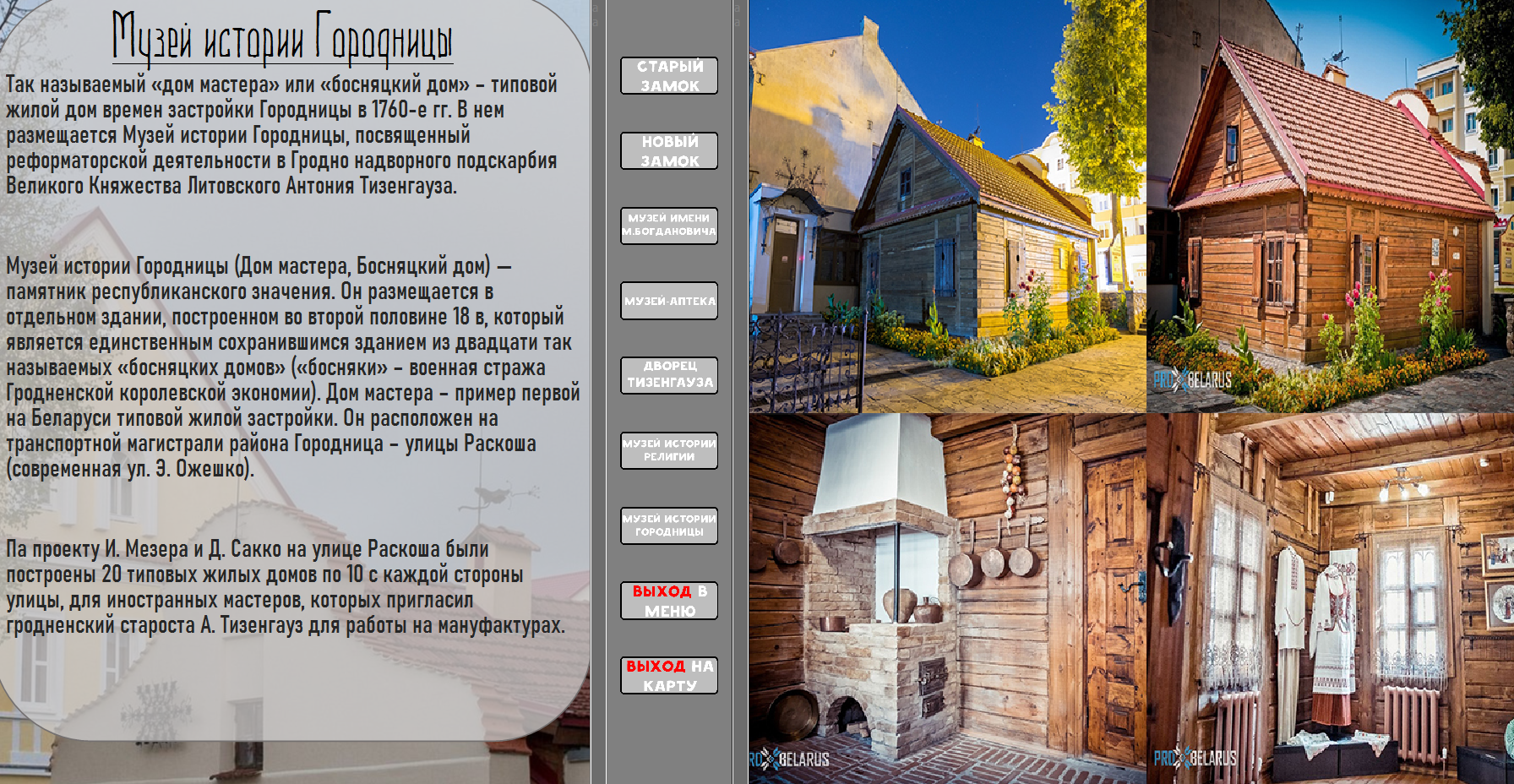


Рисунок 7 – Расположение информации на форме

При нажатии на кнопку «Настройки» пользователь имеет возможность включать или выключать музыку и звуки нажатия клавиш (рис.8,9).



Рисунок 8 – Кнопка «Настройки»



Рисунок 9 – Форма с «Настройками»

## **5.3.3 Завершение работы с программой**

Завершить работу с программой можно несколькими способами:

1. Нужно нажать на кнопку «Выход» в главном меню.
2. Также можно закрыть приложение нажав пункт меню – Выйти (Рисунок 10).

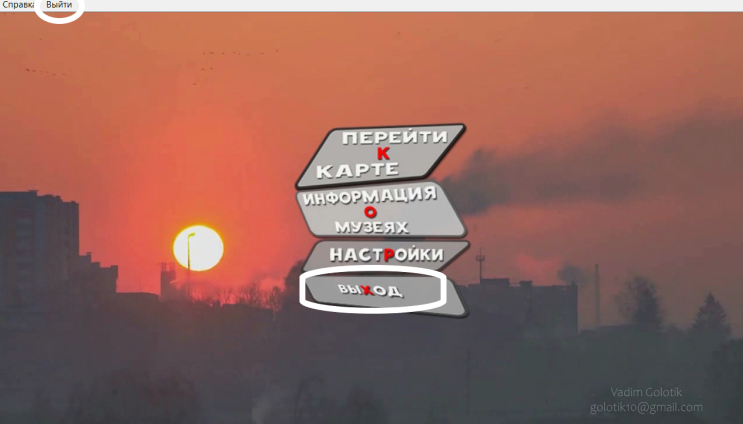


Рисунок 10 – Выход из приложения

## **5.4 Использование системы справочной информации**

Справочную систему можно запустить с помощью пункта меню «Справка» (рисунок 11)

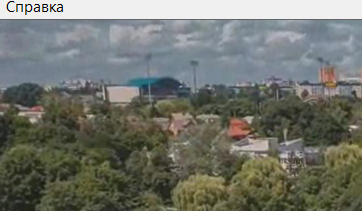


Рисунок 11 – Запуск справки

После нажатия на данный пункт меню появится форма с помощью пользователю.

## **Заключение**

Целью данного проекта было разработать познавательное приложение для большего познания об музеях город Гродно.

В поставленной задаче был реализован простой, понятный, удобный, интересный, красивый пользовательский интерфейс.

В ходе тестирования все исключительные ситуации были обработаны. Проект работает без сбоев и ошибок.

Продукт имеет дальнейшее развитие, так как на данный момент функционал недостаточно большой. Можно добавить еще много информации. Это могло бы дать началу новому многосерийному программному продукту у которого выходили бы каждый раз новые части. Несмотря на все это, данный программный продукт имеет законченный вид.

Исходя из этого, можно сделать вывод, что программа реализована успешно, а главное с душой.

## **Список использованных источников**

1. Дакетт, Д. «HTML и CSS. Разработка веб-сайтов» – М.: Изд-во «ЭКСМО.ru», – 46 с.
2. Новый замок [Электронный ресурс]. - https://ru.wikipedia.org/wiki/Новый\_замок\_(Гродно).
3. Старый замок [Электронный ресурс]. – Электронные данные. – Режим доступа: https://ru.wikipedia.org/wiki/Старый\_замок\_(Гродно).
4. Музей истории религии [Электронный ресурс]. – Электронные данные. – Режим доступа: https://ru.wikipedia.org/wiki/Гродненский\_Государственный\_музей\_религии.
5. Музей имени Максима Богдавноча [Электронный ресурс]. – Электронные данные. – Режим доступа: http://bogdanovich.grodno.museum.by/
6. Дворец Тизенгауза [Электронный ресурс]. – Электронные данные. – Режим доступа: http://www.toptur.by/country/dostoprimechatelnosti/dvorets-tyzengauza/
7. Музей Аптека Гродно [Электронный ресурс]. – Электронные данные. – Режим доступа: https://www.am.biotest.by/
8. Музей истории Городницы [Электронный ресурс]. – Электронные данные. – Режим доступа: http://gorodnica.museum.by/

## **Приложение А**

## **Листинг программы**

unit Unit1;

interface

uses

Winapi.Windows, Winapi.Messages, System.SysUtils, System.Variants, System.Classes, Vcl.Graphics,

Vcl.Controls, Vcl.Forms, Vcl.Dialogs, Vcl.ComCtrls, Vcl.StdCtrls,

Vcl.Imaging.pngimage, Vcl.ExtCtrls, Vcl.MPlayer, Vcl.Imaging.jpeg,

Vcl.Samples.Gauges;

type

TForm1 = class(TForm)

Label1: TLabel;

Label2: TLabel;

Label3: TLabel;

Label4: TLabel;

Timer1: TTimer;

MediaPlayer1: TMediaPlayer;

Gauge1: TGauge;

Image1: TImage;

Label5: TLabel;

procedure Timer1Timer(Sender: TObject);

procedure FormCreate(Sender: TObject);

private

{ Private declarations }

public

{ Public declarations }

end;

var

Form1: TForm1;

implementation

{$R \*.dfm}

uses Unit2, Unit3;

procedure TForm1.FormCreate(Sender: TObject); //при создании формы воспроизводится аудио

begin

MediaPlayer1.filename:='loadmenu.wav';

MediaPlayer1.Open;

MediaPlayer1.Play;

end;

procedure TForm1.Timer1Timer(Sender: TObject); //таймер для установки времени полосы загрузки

begin

gauge1.progress:= gauge1.progress+1;

label4.caption:='Загрузка элементов программы, подождите...' + inttostr(gauge1.progress);

if gauge1.progress=100 then

begin

Timer1.enabled:=false;

Form2.ShowModal;

Form1.close;

end;

end;

end.

unit Unit2;

interface

uses

Winapi.Windows, Winapi.Messages, System.SysUtils, System.Variants, System.Classes, Vcl.Graphics,

Vcl.Controls, Vcl.Forms, Vcl.Dialogs, Vcl.MPlayer, Vcl.ExtCtrls,

Vcl.Imaging.pngimage, mmsystem, Vcl.StdCtrls, ShellAPI, Vcl.Menus;

type

TForm2 = class(TForm)

MediaPlayer1: TMediaPlayer;

Panel1: TPanel;

Image1: TImage;

Image2: TImage;

Image3: TImage;

Image4: TImage;

MediaPlayer2: TMediaPlayer;

MainMenu1: TMainMenu;

N1: TMenuItem;

N2: TMenuItem;

N3: TMenuItem;

procedure Image1Click(Sender: TObject);

procedure Image2Click(Sender: TObject);

procedure Image4Click(Sender: TObject);

procedure FormActivate(Sender: TObject);

procedure Image3Click(Sender: TObject);

procedure N1Click(Sender: TObject);

procedure N2Click(Sender: TObject);

procedure N3Click(Sender: TObject);

private

{ Private declarations }

public

{ Public declarations }

end;

var

Form2: TForm2;

implementation

{$R \*.dfm}

uses Unit3, Unit4, Unit5, Unit6;

procedure TForm2.FormActivate(Sender: TObject); //когда форма активна вопроизводится видео

begin

MediaPlayer1.filename:='C:\Users\user\Desktop\КУРСОВОЙ ПРОЕКТ БЕРЕЗИЧ\Главное меню.wmv';

MediaPlayer1.Open;

MediaPlayer1.Play;

begin

if form4.checkbox2.checked then //если стоит соответсвующая галочка в настройках, то вопроизводится аудио

begin

MediaPlayer2.filename:='mainmenu.wav';

MediaPlayer2.Open;

MediaPlayer2.Play;

end;

end;

end;

procedure TForm2.Image1Click(Sender: TObject);//при нажатии осуществляется переход на карту

begin

Form2.hide;

form3.show;

begin

if form4.checkbox1.checked then //воспроизведение звука нажатия если в настройках стоит галочка

begin

SndPlaySound('нажатие кнопки.wav', SND\_ASYNC);//звук нажатия на кнопку

end;

end;

end;

procedure TForm2.Image2Click(Sender: TObject); // при нажатии на кнопку открывается меню настроек

begin

Form4.showmodal;

begin

if form4.checkbox1.checked then //воспроизведение звука нажатия если в настройках стоит галочка

begin

SndPlaySound('нажатие кнопки.wav', SND\_ASYNC);

end;

end;

end;

procedure TForm2.Image3Click(Sender: TObject); //при нажатии появляется форма с информацией о музяех

begin

Form5.show;

if form4.checkbox1.checked then //воспроизведение звука нажатия если в настройках стоит галочка

begin

SndPlaySound('нажатие кнопки.wav', SND\_ASYNC);

end;

end;

procedure TForm2.Image4Click(Sender: TObject); //при нажатии на кнопку программа закрывается

begin

close;

if form4.checkbox1.checked then //воспроизведение звука нажатия если в настройках стоит галочка

begin

SndPlaySound('нажатие кнопки.wav', SND\_ASYNC);

end;

end;

procedure TForm2.N1Click(Sender: TObject); //кнопка справочного меню

begin

ShellExecute(Handle, 'open', 'Help3.chm', nil,nil, SW\_NORMAL);

end;

procedure TForm2.N2Click(Sender: TObject);

begin

close;

end;

procedure TForm2.N3Click(Sender: TObject);

begin

form2.hide;

form6.show;

end;

end.

unit Unit3;

interface

uses

Winapi.Windows, Winapi.Messages, System.SysUtils, System.Variants, System.Classes, Vcl.Graphics,

Vcl.Controls, Vcl.Forms, Vcl.Dialogs, Vcl.MPlayer, Vcl.ExtCtrls,

Vcl.Imaging.pngimage, mmsystem, Vcl.StdCtrls, ShellAPI, Vcl.Menus,

Vcl.Imaging.jpeg;

type

TForm3 = class(TForm)

o: TImage;

Image1: TImage;

Image2: TImage;

Image3: TImage;

Image5: TImage;

Image6: TImage;

Image7: TImage;

Image8: TImage;

Image9: TImage;

MainMenu1: TMainMenu;

N1: TMenuItem;

N2: TMenuItem;

procedure Button1Click(Sender: TObject);

procedure Image1Click(Sender: TObject);

procedure Image8Click(Sender: TObject);

procedure Image5Click(Sender: TObject);

procedure Image9Click(Sender: TObject);

procedure Image3Click(Sender: TObject);

procedure Image2Click(Sender: TObject);

procedure Image6Click(Sender: TObject);

procedure Image7Click(Sender: TObject);

procedure N1Click(Sender: TObject);

procedure N2Click(Sender: TObject);

private

{ Private declarations }

public

{ Public declarations }

end;

var

Form3: TForm3;

implementation

{$R \*.dfm}

uses Unit2, Unit4, Unit5;

procedure TForm3.Button1Click(Sender: TObject); //при нажатии переход на главный экран

begin

Form3.hide;

Form2.show;

if form4.checkbox1.checked then

begin

SndPlaySound('нажатие кнопки.wav', SND\_ASYNC);

end;

end;

procedure TForm3.Image1Click(Sender: TObject); //при нажатии переход на форму с показом информации и фото о Старом замке

begin

Form5.show;

Form3.close;

Form5.WebBrowser1.Navigate(extractfilepath(paramstr(0))+'Старый замок.htm');

Form5.Image1.Picture.LoadFromFile ('Старый замок фото.jpg');

if form4.checkbox1.checked then //воспроизведение звука нажатия если в настройках стоит галочка

begin

SndPlaySound('нажатие кнопки.wav', SND\_ASYNC);

end;

end;

procedure TForm3.Image2Click(Sender: TObject); //при нажатии переход на форму с показом информации и фото о Музее истории религии

begin

Form3.close;

Form5.show;

Form5.WebBrowser1.Navigate(extractfilepath(paramstr(0))+'Музей истории религии.htm');

Form5.Image1.Picture.LoadFromFile ('Музей истории религии фото.jpg');

if form4.checkbox1.checked then //воспроизведение звука нажатия если в настройках стоит галочка

begin

SndPlaySound('нажатие кнопки.wav', SND\_ASYNC);

end;

end;

procedure TForm3.Image3Click(Sender: TObject);//при нажатии переход на форму с показом информации и фото о Дворце Тизенгауза

begin

Form3.close;

Form5.show;

Form5.WebBrowser1.Navigate(extractfilepath(paramstr(0))+'Дворец Тизенгауза.htm');

Form5.Image1.Picture.LoadFromFile ('Дворец Тизенгауза фото.jpg');

if form4.checkbox1.checked then //воспроизведение звука нажатия если в настройках стоит галочка

begin

SndPlaySound('нажатие кнопки.wav', SND\_ASYNC);

end;

end;

procedure TForm3.Image5Click(Sender: TObject); //при нажатии переход на форму с показом информации и фото о Музее Богдановича

begin

form5.show;

form3.close;

Form5.WebBrowser1.Navigate(extractfilepath(paramstr(0))+'Музей Богдановича.htm');

Form5.Image1.Picture.LoadFromFile('фото.jpg');

if form4.checkbox1.checked then //воспроизведение звука нажатия если в настройках стоит галочка

begin

SndPlaySound('нажатие кнопки.wav', SND\_ASYNC);

end;

begin

end;

end;

procedure TForm3.Image6Click(Sender: TObject);//при нажатии переход на форму с показом информации и фото о Аптеке-музее

begin

form3.close;

form5.show;

Form5.WebBrowser1.Navigate(extractfilepath(paramstr(0))+'Аптека-музей.htm');

Form5.Image1.Picture.LoadFromFile ('Аптека-музей.jpg');

if form4.checkbox1.checked then //воспроизведение звука нажатия если в настройках стоит галочка

begin

SndPlaySound('нажатие кнопки.wav', SND\_ASYNC);

end;

end;

procedure TForm3.Image7Click(Sender: TObject);//при нажатии переход на форму с показом информации и фото о музее истории Городницы

begin

Form3.close;

Form5.show;

Form5.WebBrowser1.Navigate(extractfilepath(paramstr(0))+'Музей истории Городницы.htm');

Form5.Image1.Picture.LoadFromFile ('Музей истории Городницы фото.jpg');

if form4.checkbox1.checked then //воспроизведение звука нажатия если в настройках стоит галочка

begin

SndPlaySound('нажатие кнопки.wav', SND\_ASYNC);

end;

end;

procedure TForm3.Image8Click(Sender: TObject); //при нажатии переход на форму с показом информации и фото о Новом замке

begin

form5.show;

form3.close;

Form5.WebBrowser1.Navigate(extractfilepath(paramstr(0))+'Новый замок.htm');

Form5.Image1.Picture.LoadFromFile ('Новый замок фото.jpg');

if form4.checkbox1.checked then //воспроизведение звука нажатия если в настройках стоит галочка

begin

SndPlaySound('нажатие кнопки.wav', SND\_ASYNC);

end;

end;

procedure TForm3.Image9Click(Sender: TObject); //переход на главное меню

begin

Form3.hide;

Form2.show;

if form4.checkbox1.checked then

begin

SndPlaySound('нажатие кнопки.wav', SND\_ASYNC);//звук нажатия кнопки

end;

end;

procedure TForm3.N1Click(Sender: TObject);

begin

ShellExecute(Handle, 'open', 'Help3.chm', nil,nil, SW\_NORMAL);

end;

procedure TForm3.N2Click(Sender: TObject);

begin

close;

end;

end.

unit Unit4;

interface

uses

Winapi.Windows, Winapi.Messages, System.SysUtils, System.Variants, System.Classes, Vcl.Graphics,

Vcl.Controls, Vcl.Forms, Vcl.Dialogs, Vcl.StdCtrls, mmsystem, IniFiles,

Vcl.ExtCtrls, Vcl.Imaging.pngimage,SHDocVw, Vcl.Menus, ShellAPI;

type

TForm4 = class(TForm)

CheckBox1: TCheckBox;

Image1: TImage;

CheckBox2: TCheckBox;

procedure Button1Click(Sender: TObject);

procedure FormCreate(Sender: TObject);

procedure Image1Click(Sender: TObject);

procedure Button2Click(Sender: TObject);

procedure N1Click(Sender: TObject);

private

{ Private declarations }

public

{ Public declarations }

end;

var

Form4: TForm4;

Ini: TIniFile;

implementation

{$R \*.dfm}

uses Unit2;

procedure TForm4.Button1Click(Sender: TObject); //при постановке флажка в том или ином пункте сохраняются настройки

begin

Ini:= TIniFile.create(Extractfilepath(Paramstr(0))+'config.ini');

Ini.WriteBool('Settings','Sound', checkbox1.checked); //измена настроек в config.ini

if form4.checkbox1.checked then

begin

SndPlaySound('нажатие кнопки.wav', SND\_ASYNC);//звук при нажатии на кнопки

begin

Ini:= TIniFile.create(Extractfilepath(Paramstr(0))+'config.ini');

Ini.WriteBool('Settings','Sound2', checkbox2.checked);

if form4.checkbox2.checked then

begin

SndPlaySound('mainmenu.wav', SND\_ASYNC);

end;

end;

end;

end;

procedure TForm4.Button2Click(Sender: TObject);

begin

WinExec('Help\Help.chm', SW\_MAXIMIZE);

end;

procedure TForm4.FormCreate(Sender: TObject);

begin

Ini:= TIniFile.create(Extractfilepath(Paramstr(0))+'config.ini');

checkbox1.checked:=Ini.ReadBool('Settings','Sound', checkbox1.checked);

begin

Ini:= TIniFile.create(Extractfilepath(Paramstr(0))+'config.ini');

checkbox2.checked:=Ini.ReadBool('Settings','Sound2', checkbox2.checked);

end;

end;

procedure TForm4.Image1Click(Sender: TObject);

begin

Ini:= TIniFile.create(Extractfilepath(Paramstr(0))+'config.ini');

Ini.WriteBool('Settings','Sound', checkbox1.checked);

if form4.checkbox1.checked then

begin

SndPlaySound('нажатие кнопки.wav', SND\_ASYNC);

begin

form4.close;

begin

Ini:= TIniFile.create(Extractfilepath(Paramstr(0))+'config.ini');

Ini.WriteBool('Settings','Sound2', checkbox2.checked);

if form4.checkbox2.checked then

begin

SndPlaySound('mainmenu.wav', SND\_ASYNC);

end;

end;

end;

end;

end;

procedure TForm4.N1Click(Sender: TObject);

begin

ShellExecute(0,Pchar('Open'),PChar('Help.chm'),nil,nil,SW\_SHOW);

end;

end.

unit Unit5;

interface

uses

Winapi.Windows, Winapi.Messages, System.SysUtils, System.Variants, System.Classes, Vcl.Graphics,

Vcl.Controls, Vcl.Forms, Vcl.Dialogs, Vcl.StdCtrls, Vcl.OleCtrls, SHDocVw,

Vcl.ExtCtrls, Vcl.Imaging.jpeg, Vcl.Imaging.pngimage, Vcl.Menus, ShellAPI,mmsystem;

type

TForm5 = class(TForm)

WebBrowser1: TWebBrowser;

Image1: TImage;

Image3: TImage;

Image4: TImage;

Image5: TImage;

Image6: TImage;

Image8: TImage;

Image9: TImage;

Image10: TImage;

Image11: TImage;

Image12: TImage;

aa: TMemo;

Memo1: TMemo;

MainMenu1: TMainMenu;

N1: TMenuItem;

N2: TMenuItem;

procedure Button1Click(Sender: TObject);

procedure FormCreate(Sender: TObject);

procedure Button2Click(Sender: TObject);

procedure Button3Click(Sender: TObject);

procedure Image12Click(Sender: TObject);

procedure Image11Click(Sender: TObject);

procedure Image3Click(Sender: TObject);

procedure Image4Click(Sender: TObject);

procedure Image5Click(Sender: TObject);

procedure Image8Click(Sender: TObject);

procedure Image9Click(Sender: TObject);

procedure Image6Click(Sender: TObject);

procedure Image10Click(Sender: TObject);

procedure N1Click(Sender: TObject);

procedure N2Click(Sender: TObject);

private

{ Private declarations }

public

{ Public declarations }

end;

var

Form5: TForm5;

implementation

{$R \*.dfm}

uses Unit3, Unit4, Unit2;

procedure TForm5.Button1Click(Sender: TObject); //Вернутся на карту

begin

Form5.hide;

Form3.show;

end;

procedure TForm5.Button2Click(Sender: TObject); //показ информации о новом замке

begin

WebBrowser1.Navigate(extractfilepath(paramstr(0))+'Новый замок.htm');

end;

procedure TForm5.Button3Click(Sender: TObject);//показ информации и фото о музее Богдановича

begin

WebBrowser1.Navigate(extractfilepath(paramstr(0))+'Музей Богдановича.htm');

end;

procedure TForm5.FormCreate(Sender: TObject); //показ информации и фото о старом замке

begin

WebBrowser1.Navigate(extractfilepath(paramstr(0))+'Старый замок.htm');

Form5.Image1.Picture.LoadFromFile ('Старый замок фото.jpg');

end;

procedure TForm5.Image10Click(Sender: TObject);//показ информации и фото о музее истории Городницы

begin

Form5.WebBrowser1.Navigate(extractfilepath(paramstr(0))+'Музей истории Городницы.htm');

Form5.Image1.Picture.LoadFromFile ('Музей истории Городницы фото.jpg');

if form4.checkbox1.checked then //воспроизведение звука нажатия если в настройках стоит галочка

begin

SndPlaySound('нажатие кнопки.wav', SND\_ASYNC);

end;

end;

procedure TForm5.Image11Click(Sender: TObject);//переход на главное меню

begin

form5.close;

form2.show;

if form4.checkbox1.checked then //воспроизведение звука нажатия если в настройках стоит галочка

begin

SndPlaySound('нажатие кнопки.wav', SND\_ASYNC);

end;

end;

procedure TForm5.Image12Click(Sender: TObject); //переход на карту

begin

form5.close;

form3.show;

if form4.checkbox1.checked then //воспроизведение звука нажатия если в настройках стоит галочка

begin

SndPlaySound('нажатие кнопки.wav', SND\_ASYNC);

end;

end;

procedure TForm5.Image3Click(Sender: TObject); //показ информации и фото о Старом музее

begin

Form5.WebBrowser1.Navigate(extractfilepath(paramstr(0))+'Старый замок.htm');

Form5.Image1.Picture.LoadFromFile ('Старый замок фото.jpg');

if form4.checkbox1.checked then //воспроизведение звука нажатия если в настройках стоит галочка

begin

SndPlaySound('нажатие кнопки.wav', SND\_ASYNC);

end;

end;

procedure TForm5.Image4Click(Sender: TObject); //показ информации и фото о Новом замке

begin

Form5.WebBrowser1.Navigate(extractfilepath(paramstr(0))+'Новый замок.htm');

Form5.Image1.Picture.LoadFromFile ('Новый замок фото.jpg');

if form4.checkbox1.checked then //воспроизведение звука нажатия если в настройках стоит галочка

begin

SndPlaySound('нажатие кнопки.wav', SND\_ASYNC);

end;

end;

procedure TForm5.Image5Click(Sender: TObject);//показ информации и фото о музее Богдановича

begin

Form5.WebBrowser1.Navigate(extractfilepath(paramstr(0))+'Музей Богдановича.htm');

Form5.Image1.Picture.LoadFromFile('фото.jpg');

if form4.checkbox1.checked then //воспроизведение звука нажатия если в настройках стоит галочка

begin

SndPlaySound('нажатие кнопки.wav', SND\_ASYNC);

end;

end;

procedure TForm5.Image6Click(Sender: TObject); //показ информации и фото о аптеке-музее

begin

Form5.WebBrowser1.Navigate(extractfilepath(paramstr(0))+'Аптека-музей.htm');

Form5.Image1.Picture.LoadFromFile ('Аптека-музей.jpg');

if form4.checkbox1.checked then //воспроизведение звука нажатия если в настройках стоит галочка

begin

SndPlaySound('нажатие кнопки.wav', SND\_ASYNC);

end;

end;

procedure TForm5.Image8Click(Sender: TObject);//показ информации и фото о дворце Тизенгауза

begin

Form5.WebBrowser1.Navigate(extractfilepath(paramstr(0))+'Дворец Тизенгауза.htm');

Form5.Image1.Picture.LoadFromFile ('Дворец Тизенгауза фото.jpg');

if form4.checkbox1.checked then //воспроизведение звука нажатия если в настройках стоит галочка

begin

SndPlaySound('нажатие кнопки.wav', SND\_ASYNC);

end;

end;

procedure TForm5.Image9Click(Sender: TObject); //показ информации и фото о музее истории религии

begin

Form5.WebBrowser1.Navigate(extractfilepath(paramstr(0))+'Музей истории религии.htm');

Form5.Image1.Picture.LoadFromFile ('Музей истории религии фото.jpg');

if form4.checkbox1.checked then //воспроизведение звука нажатия если в настройках стоит галочка

begin

SndPlaySound('нажатие кнопки.wav', SND\_ASYNC);

end;

end;

procedure TForm5.N1Click(Sender: TObject);

begin

ShellExecute(Handle, 'open', 'Help3.chm', nil,nil, SW\_NORMAL);

end;

procedure TForm5.N2Click(Sender: TObject);

begin

close;

end;

end.

unit Unit6;

interface

uses

Winapi.Windows, Winapi.Messages, System.SysUtils, System.Variants, System.Classes, Vcl.Graphics,

Vcl.Controls, Vcl.Forms, Vcl.Dialogs, Vcl.StdCtrls, Vcl.ExtCtrls,

Vcl.Imaging.pngimage, Vcl.Imaging.jpeg,mmsystem;

type

TForm6 = class(TForm)

Label1: TLabel;

RadioGroup1: TRadioGroup;

RadioGroup2: TRadioGroup;

Image1: TImage;

Image2: TImage;

Label2: TLabel;

Image3: TImage;

procedure Image1Click(Sender: TObject);

procedure RadioGroup1Click(Sender: TObject);

procedure Image3Click(Sender: TObject);

private

{ Private declarations }

public

{ Public declarations }

end;

var

Form6: TForm6;

f: text;

s: string;

Nvern, ball: integer;

implementation

{$R \*.dfm}

uses Unit4, Unit2;

procedure TForm6.Image1Click(Sender: TObject);

begin

if form4.checkbox1.checked then //воспроизведение звука нажатия если в настройках стоит галочка

begin

SndPlaySound('нажатие кнопки.wav', SND\_ASYNC);

end;

//Если выбран вариант ответа и не достигнут конец файла

if (RadioGroup2.ItemIndex>-1) and (not Eof(f)) then begin

if RadioGroup2.ItemIndex = Nvern-1 then ball:=ball+1; //Если выбранный вариант соответствует

RadioGroup2.Items.Clear; //номеру верного ответа то балл прибавляется

Repeat //и очищается поле для следующего вопроса

if (s[1]='-') then begin

delete(s,1,1);

RadioGroup2.Caption:=s;

end

else if s[1]='\*' then begin

delete(s,1,1);

Nvern:=StrToInt(s);

end

else RadioGroup2.Items.Add(s);

readln(f,s);

Label1.Caption:=('Следующий вопрос: ' +s);

until (s[1]='-') or (Eof(f));

end

//Если конец файла достигнут, значит вопросы закончились

Else if Eof(f) then begin

delete(s,1,1);

Nvern:=StrToInt(s);

if RadioGroup2.ItemIndex = Nvern-1 then ball:=ball+1;

Label1.Caption:=IntToStr(ball);

Label2.Visible:=True; //Вывод количества баллов

CloseFile(f);

Image1.Enabled:=False; //кнопка становится недоступной в завершении проекта

end;

end;

procedure TForm6.Image3Click(Sender: TObject);

begin

form6.close;

form2.show;

if form4.checkbox1.checked then //воспроизведение звука нажатия если в настройках стоит галочка

begin

SndPlaySound('нажатие кнопки.wav', SND\_ASYNC);

end;

end;

procedure TForm6.RadioGroup1Click(Sender: TObject);

begin

RadioGroup1.Enabled:=false; //Выбор варианта становится недоступен

RadioGroup2.Enabled:=true; //Доступным становится поле с вопросом

Image1.Enabled:=true; //Кнопка Далее

case RadioGroup1.ItemIndex of //В зависимости от выбранного варианта переменная f

0: AssignFile( f ,'вариант1.txt', CP\_UTF8);//связывается с разными файлами

1: AssignFile( f ,'вариант2.txt', CP\_UTF8);

end;

reset(f); //Открываем файл для чтения

readln(f,s); //Считываем первую строку из файла

ball:=0; //изначально количество баллов 0

repeat

if (s[1]='-') then begin //Если первый символ строки ‘-‘ значит это вопрос

delete(s,1,1);

RadioGroup2.Caption:=s;

end

else if s[1]='\*' then

begin //Если перв символ ‘\*’ значит это номер верного ответа

delete(s,1,1);

Nvern:=StrToInt(s);

end

else RadioGroup2.Items.Add(s); //Иначе это вариант ответа

readln(f,s); //Считываем следующую строку из файла

until (s[1]='-') or (Eof(f)); //Считывание и отправление вариантов ответов в RadiGroup до тех пор

// пока не достигнут следующий вопрос или конец файла

end;

end.