%3CmxGraphModel%3E%3Croot%3E%3CmxCell%20id%3D%220%22%2F%3E%3CmxCell%20id%3D%221%22%20parent%3D%220%22%2F%3E%3CmxCell%20id%3D%222%22%20value%3D%22%26lt%3Bfont%20style%3D%26quot%3Bfont-size%3A%2030px%26quot%3B%26gt%3B%D0%BF%D1%80%D0%BE%D1%81%D0%BC%D0%BE%D1%82%D1%80%20%D0%BE%D0%BA%D0%BD%D0%B0%20%D0%BE%D0%B1%20%D0%B0%D0%B2%D1%82%D0%BE%D1%80%D0%B5%26lt%3B%2Ffont%26gt%3B%22%20style%3D%22rounded%3D0%3BwhiteSpace%3Dwrap%3Bhtml%3D1%3B%22%20vertex%3D%221%22%20parent%3D%221%22%3E%3CmxGeometry%20x%3D%224390%22%20y%3D%22540%22%20width%3D%22329%22%20height%3D%22130%22%20as%3D%22geometry%22%2F%3E%3C%2FmxCell%3E%3C%2Froot%3E%3C%2FmxGraphModel%3E%3CmxGraphModel%3E%3Croot%3E%3CmxCell%20id%3D%220%22%2F%3E%3CmxCell%20id%3D%221%22%20parent%3D%220%22%2F%3E%3CmxCell%20id%3D%222%22%20value%3D%22%26lt%3Bfont%20style%3D%26quot%3Bfont-size%3A%2030px%26quot%3B%26gt%3B%D0%BF%D1%80%D0%BE%D1%81%D0%BC%D0%BE%D1%82%D1%80%20%D0%BE%D0%BA%D0%BD%D0%B0%20%D0%BE%D0%B1%20%D0%B0%D0%B2%D1%82%D0%BE%D1%80%D0%B5%26lt%3B%2Ffont%26gt%3B%22%20style%3D%22rounded%3D0%3BwhiteSpace%3Dwrap%3Bhtml%3D1%3B%22%20vertex%3D%221%22%20parent%3D%221%22%3E%3CmxGeometry%20x%3D%224390%22%20y%3D%22540%22%20width%3D%22329%22%20height%3D%22130%22%20as%3D%22geometry%22%2F%3E%3C%2FmxCell%3E%3C%2Froot%3E%3C%2FmxGraphModel%3E%3CmxGraphModel%3E%3Croot%3E%3CmxCell%20id%3D%220%22%2F%3E%3CmxCell%20id%3D%221%22%20parent%3D%220%22%2F%3E%3CmxCell%20id%3D%222%22%20value%3D%22%26lt%3Bfont%20style%3D%26quot%3Bfont-size%3A%2030px%26quot%3B%26gt%3B%D0%BF%D1%80%D0%BE%D1%81%D0%BC%D0%BE%D1%82%D1%80%20%D0%BE%D0%BA%D0%BD%D0%B0%20%D0%BE%D0%B1%20%D0%B0%D0%B2%D1%82%D0%BE%D1%80%D0%B5%26lt%3B%2Ffont%26gt%3B%22%20style%3D%22rounded%3D0%3BwhiteSpace%3Dwrap%3Bhtml%3D1%3B%22%20vertex%3D%221%22%20parent%3D%221%22%3E%3CmxGeometry%20x%3D%224390%22%20y%3D%22540%22%20width%3D%22329%22%20height%3D%22130%22%20as%3D%22geometry%22%2F%3E%3C%2FmxCell%3E%3C%2Froot%3E%3C%2FmxGraphModel%3E**Введение**

На курсовое проектирование была поставлена задача разработать электронное средство обучения на тему «Проведение фотосессии».

Цель курсового проекта заключается в ознакомлении с видами фотосессий, обучением организации и проведения разных видов фотосессий.

Создаваемое электронное средство обучения будет рассчитано на любого рода пользователей. Применить его смогут не только люди, владеющими правилами проведения фотосессий и имеющие опыт работы с фотоаппаратом, но и просто люди, заинтересовавшиеся данной темой.

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 создаётся 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.

%3CmxGraphModel%3E%3Croot%3E%3CmxCell%20id%3D%220%22%2F%3E%3CmxCell%20id%3D%221%22%20parent%3D%220%22%2F%3E%3CmxCell%20id%3D%222%22%20value%3D%22%26lt%3Bfont%20style%3D%26quot%3Bfont-size%3A%2030px%26quot%3B%26gt%3B%D0%BF%D1%80%D0%BE%D1%81%D0%BC%D0%BE%D1%82%D1%80%20%D0%BE%D0%BA%D0%BD%D0%B0%20%D0%BE%D0%B1%20%D0%B0%D0%B2%D1%82%D0%BE%D1%80%D0%B5%26lt%3B%2Ffont%26gt%3B%22%20style%3D%22rounded%3D0%3BwhiteSpace%3Dwrap%3Bhtml%3D1%3B%22%20vertex%3D%221%22%20parent%3D%221%22%3E%3CmxGeometry%20x%3D%224390%22%20y%3D%22540%22%20width%3D%22329%22%20height%3D%22130%22%20as%3D%22geometry%22%2F%3E%3C%2FmxCell%3E%3C%2Froot%3E%3C%2FmxGraphModel%3E

**1 Анализ задачи**

* 1. **Постановка задачи**

**Наименование задачи:** разработка электронного средства обучения «Проведение фотосессии».

**Цель разработки:** в ознакомлении с видами фотосессий, обучением организации и проведения разных видов фотосессий

**Назначение:** данное электронное средство обучения разрабатывается для людей любого возраста, желающих пройти данный курс обучения впервые или для закрепления материала.

**Периодичность использования:** зависит от нужд потребителя, может использоваться ежедневно.

**Источники и способы получения данных:**

К входной информации можно отнести вводимые пользователем значения, например, ответы на вопросы при прохождении одного из тестов.

К выходной – результат прохождения теста, проведение фотосессии, основные компоненты фотоаппарата , используемы режимы работы с объектами.

Постоянной информацией в проекте будут являться текстовые файлы, графические изображения, видео информация и др.

Разрабатываемый программный продукт должен позволять выполнить следующие действия:

- просмотр теории по данным разделам;

- прохождение теста по теме: «Проведение фотосессии+компоненты фотоаппарата»;

- просмотр видеоматериала «Как правильно настроить фотоаппарат».

Для разработки данного проекта будет выбрана среда Delphi (Rad Studio 11.1) так как это одна из мощнейших сред для разработки приложений такого уровня. Delphi — это среда объектно-ориентированного программирования, относящаяся к классу RAD – (Rapid Application Development – «Средство быстрой разработки приложений»), основанная на языке Object Pascal. Используется для разработки и поддержки программного обеспечения.

При разработке данного программного продукта был использован компьютер со следующими характеристиками:

- Процессор: Intel(R) Pentium(R) Silver N5000;

- ОЗУ: 4Gb;

- Память: SSD 1028Gb;

- ОС – Windows 10.

Также для разработки программы необходимы:

- Google-браузер – нужен для нахождения информации и картинок;

- WPS Office – нужен для написания пояснительной записки;

- WPS Office – нужен для создания отчётной презентации;

- Help + Manual 8 – создание справочной системы;

* 1. **Требования к приложению**

На этапе исследования предметной области был установленный целый ряд требований, которые предъявляются к разрабатываемой задаче. При моделировании форм следует учесть такие моменты:

- интуитивно понятный интерфейс;

- небольшое количество информации на каждой форме;

- небольшие окна формы;

- визуально понятный и приятный глазу интерфейс;

- небольшие кнопки.

Минимальные системные требования:

- 1.8 GHz процессор (или более быстрый);

- 1 GB RAM;

- от 300 MB доступного места на жестком диске.

Рекомендуемые:

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

- 3 GB RAM;

- от 500 MB доступного места на жестком диске.

**Требования к интерфейсу:** при разработке приложения должны быть использованы преимущественного строгие оттенки . Основные разделы приложения должны быть доступны с первой страницы. Грамотный пользовательский интерфейс.

Следовательно, каждое окно должно иметь ясную визуальную иерархию своих элементов. Фрагменты текста должны располагаться на экране так, чтобы пользователю было просто и понятно принимать информацию.

Пользователь не должен испытывать какого-либо дискомфорта в плане восприятия информация, отображённой на экране. Объекты (рисунки и символы) не должны быть слишком мелкие.

Отказы программы вследствие некорректных действий пользователя при взаимодействии с программой через графический интерфейс не должны влиять на конечный результат.

**2 Проектирование задачи**

**2.1 Организация данных**

Проектирование задачи – это очень важный и ответственный этап в разработке любого приложения.

Важным является он вследствие того, что методы, по средствам которых пользователь управляет формами, построены на высокой степени специализации каждого из компонентов.

Необходимым условием при разработке данного приложения является описание организации данных, т.е. логическая и физическая структура данных в контексте среды разработки. В разрабатываемой программе будут использоваться один вид данных.

Первым видом являются данные, которые будут введены разработчиком на этапе реализации задачи. Сюда можно отнести изображения (иконки), описание, видеоизображение.

Описание данных – теоретическая текстовая информация. Использование изображений для демонстрации процесса создания, проведения фотосессии.

Представление изображений графических работ, видеоизображения.

Система справочной информации будет представлена файлом справки, которая будет содержать информацию об функционале приложения, правилах ее использования.

Таким образом, организация данных является важной задачей при разработке данной и любой программы.

**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 ее задачей будет являться: обучение пользователей основам фотосессии .

Основными функциями приложения являются:

- Научить пользователя пользоваться фотоаппаратом;

- Просмотр видеоматериала по настройке фотоаппарата;

- Просмотр примеров авторских работ и не только ;

- Прохождение теста «Проведение фотосессии + настройка фотоаппарата»;

-Изучение справочной информации про функционал программы .

**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. Во-первых, особое внимание следует уделить интерфейсу. Разработчик должен так организовать внешний вид своей программы, чтобы пользователь понял, как работать с данной программой.

Для организации эффективной 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. Приложение должно позволить пользователю решать задачи, затрачивая значительно меньше усилий, чем при работе с разрозненными объектами.

Ниже представлена система меню и организация навигации между окнами программы (Рисунок 1):

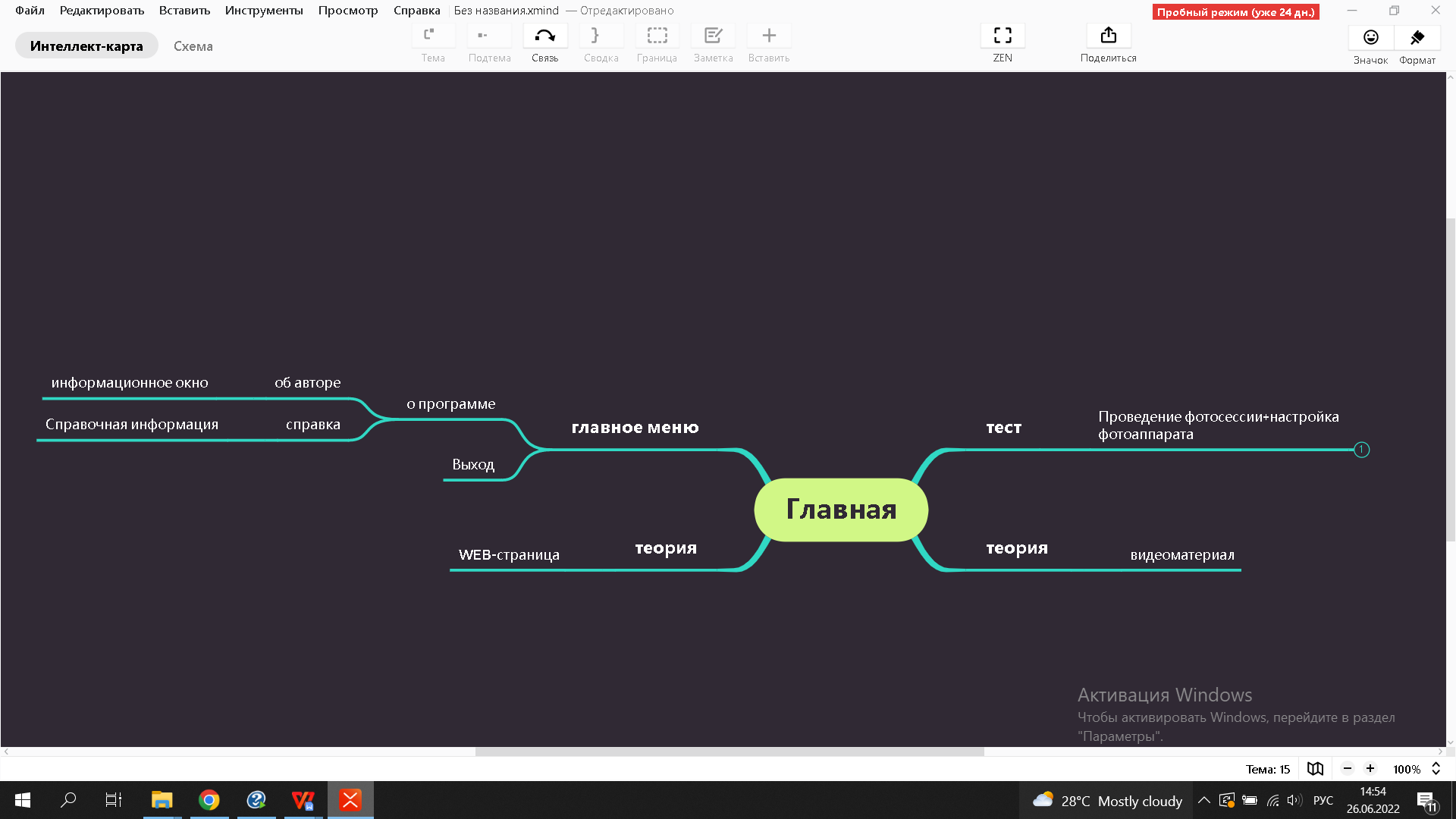


Рисунок 1 – Навигация между окнами программы

**Прототип** – это наглядная модель пользовательского интерфейса. В сущности это «черновик», созданный на основе представления разработчика о потребностях пользователя. **Итоговое отображение программы может отличаться от прототипа.** Ниже можно наблюдать прототипы:

1. Главного окна (Рисунок 2)

2. Окна, содержащее основную информацию (Рисунок 3)

3. Окна, содержащий тест (Рисунок 4)

4. Окно, содержащее видеоизображение (Рисунок 5)

5.Окна, загрузки (Рисунок 6)

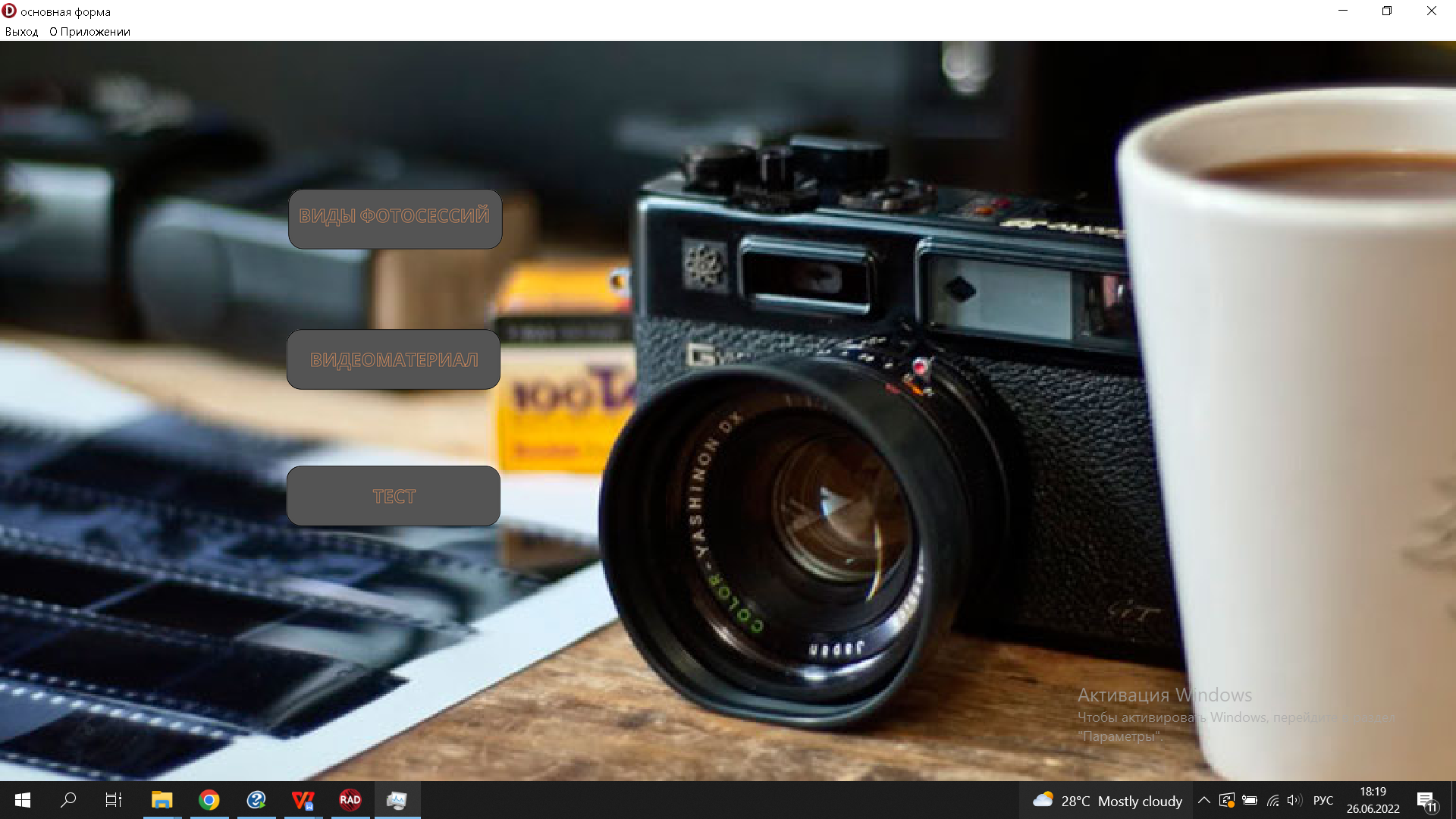


Рисунок 2 – Прототип главного окна.



Рисунок 3 – Прототип окна, содержащее основную

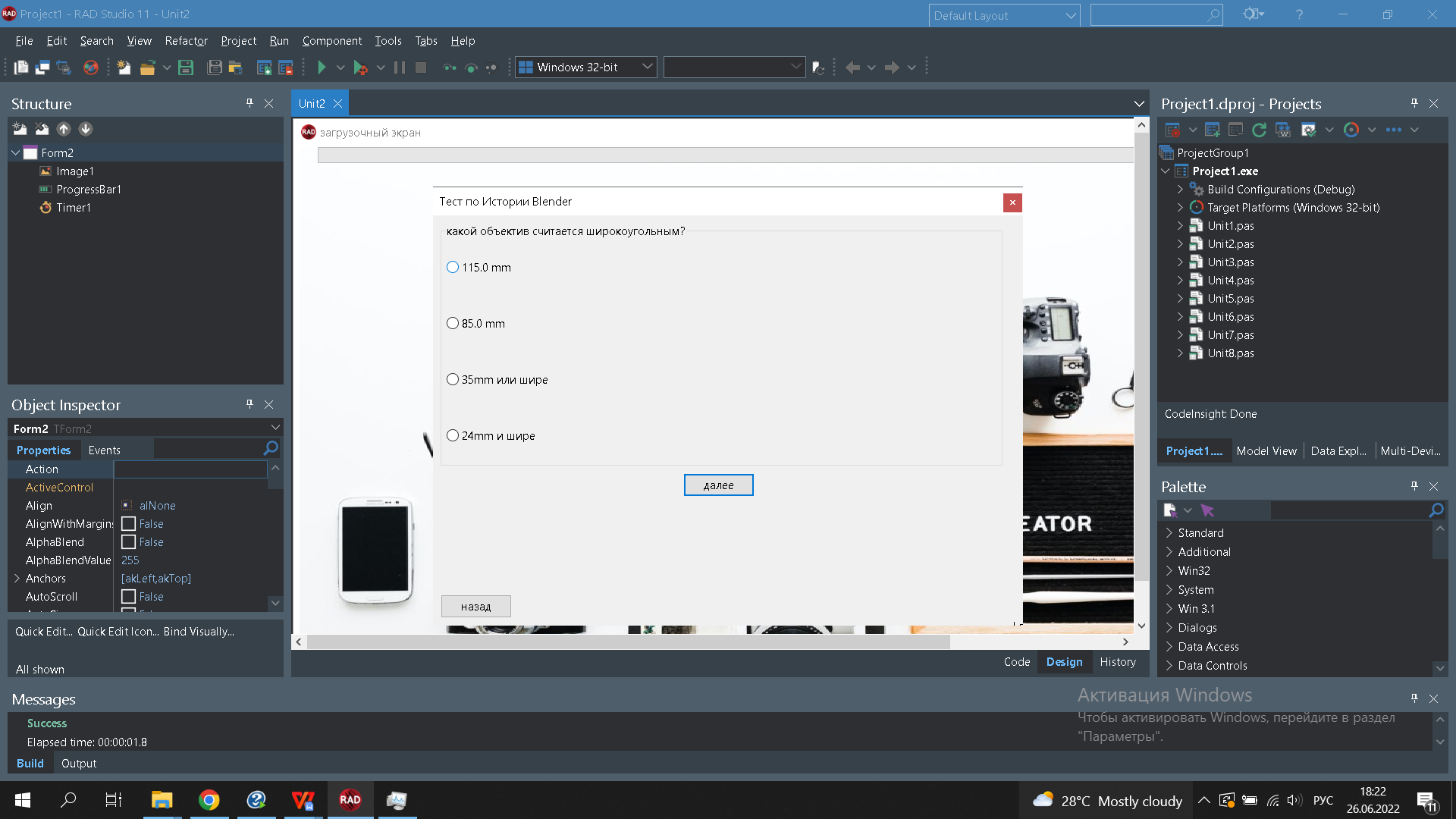


Рисунок 4 - Прототип окна, содержащий тест

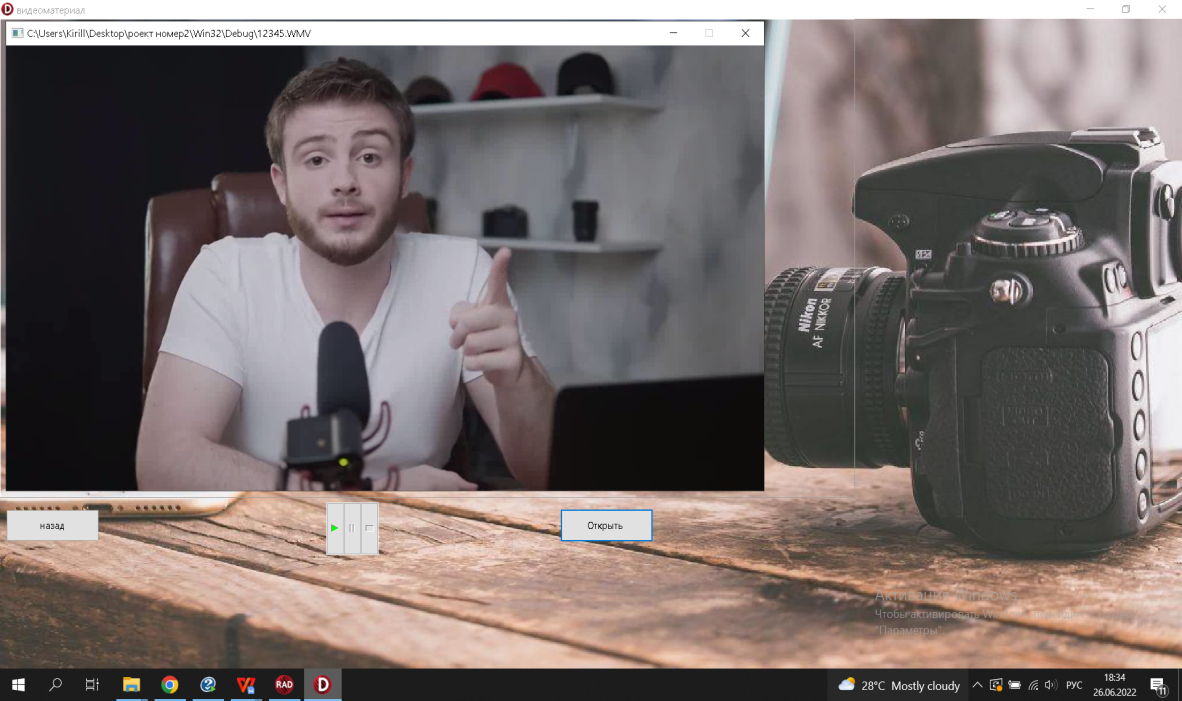


Рисунок 5- Прототип окна, содержащее видеоизображение

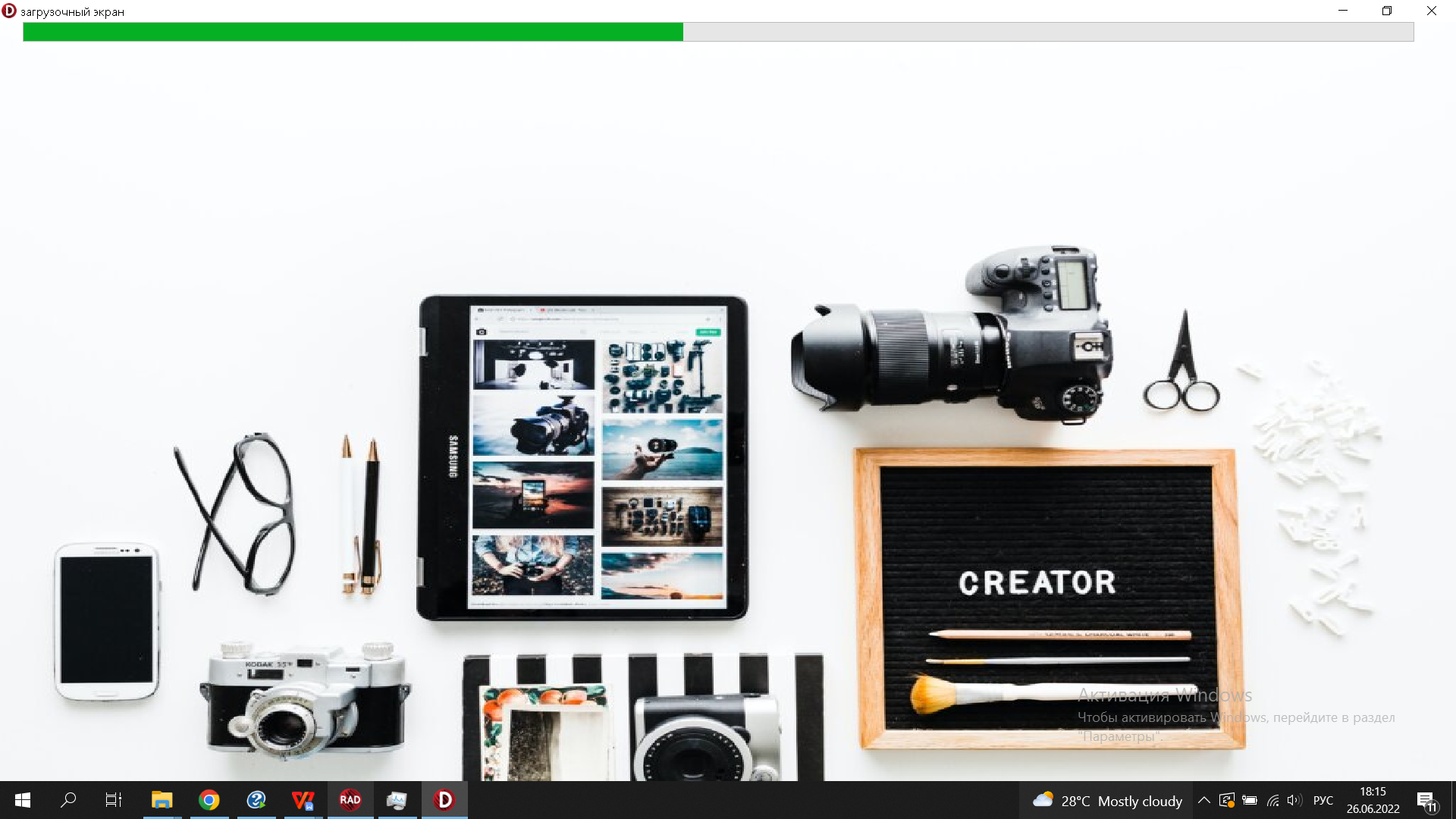
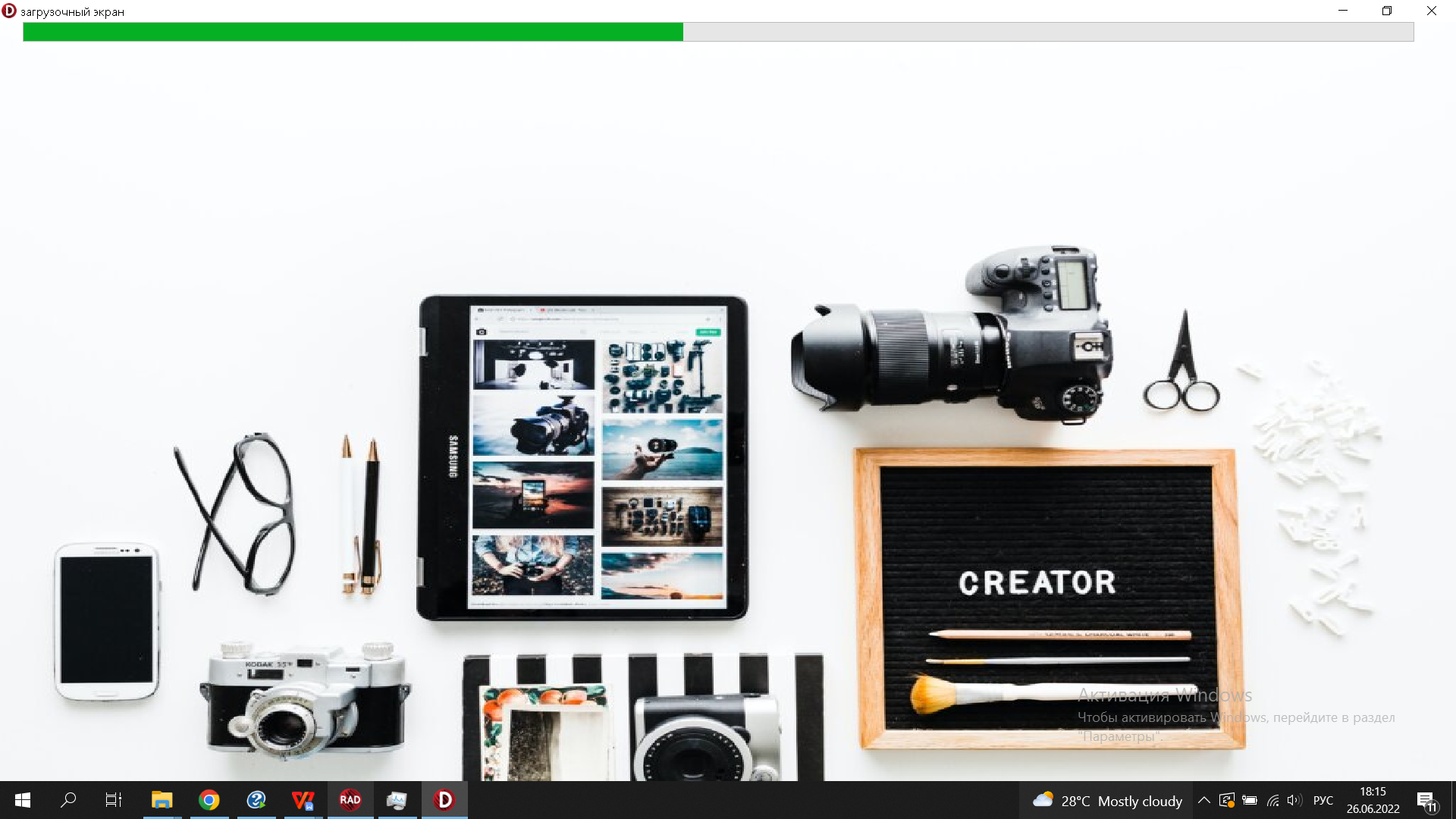


Рисунок 6 – Прототип окна, загрузки

Таким образом, для успешной работы всего проекта в целом следует непременно тесно связать все эти компоненты и заставить их работать.

**3. Реализация**

**3.1 Реализация проекта**

**3.1.1 Структура программы**

Данный курсовой проект содержит 8 модулей. Далее будет описано назначение каждого из них.

Модуль Main является модулем, за которым закреплена главная форма. C которой можно перейти на любую другую форму проекта.

Модуль Ther - модуль, содержащий теоретический материал тема проекта. На нем можно прочитать теорию про моделирование в Blender, историю Blender.

Модуль Port - модуль, содержащий примеры авторских работ. На ней можно просмотреть работы автора проекта.

Модуль Test - модуль, содержащий тестовую часть проекта. На ней можно пройти два теста, созданных автором.

Модуль ChoiseTest – модуль, содержащий форму для выбора одного из тестов.

Модуль Avt – модуль, содержащий информацию об авторе проекта. На форме можно прочить информацию об авторе.

Модуль dpictur – модуль, содержащий форму с увеличенным изображением. На ней можно увидеть изображение в большем размере.

Модуль Video – модуль содержащий видеоизображение. На ней можно просмотреть авторский видео фрагмент.

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

Описание разработанных процедур и функций приводится в таблице 1.

Таблица 1 - Процедуры и функции

|  |  |  |  |
| --- | --- | --- | --- |
| Имя процедуры (функции) | В каком модуле находится | За каким элементом управления закреплена | Назначение |
| 2 | 3 | 4 | 5 |
| procedure TMainForm.Image3Click(Sender: TObject); | Main | Image3 | Кнопка «Теория».Переход на форму с теорией |
| procedure TMainForm.Image4Click(Sender: TObject); | Main | Image4 | Кнопка «Портфолио» переход на форму с портфолио |
| procedure TTheory.Image3Click(Sender: TObject); | Ther | Image3 | Кнопка «Вперед» |
| procedure TTheory.Image4Click(Sender: TObject); | Ther | Image4 | Кнопка «Назад» |
| procedure TPortf.Image11DblClick(Sender: TObject); | Port | Image11 | Открытие картинки в большой форме |
| procedure TPortf.Image11MouseEnter(Sender: TObject); | Port | Image11 | Появление надписи «Press me» |
| procedure TPortf.Image11MouseLeave(Sender: TObject); | Port | Image11 | Исчезновение надписи «Press me» |
| procedure TPortf.Image12DblClick(Sender: TObject); | Port | Image12 | Открытие картинки в большой форме |
| procedure TPortf.Image12MouseEnter(Sender: TObject); | Port | Image12 | Появление надписи «Press me» |
| procedure TPortf.Image12MouseLeave(Sender: TObject); | Port | Image12 | Исчезновение надписи «Press me» |
| procedure TPortf.Image13DblClick(Sender: TObject); | Port | Image13 | Открытие картинки в большой форме |
| procedure TPortf.Image13MouseEnter(Sender: TObject); | Port | Image13 | Появление надписи «Press me» |
| procedure TPortf.Image13MouseLeave(Sender: TObject); | Port | Image13 | Исчезновение надписи «Press me» |
| procedure TPortf.Image14DblClick(Sender: TObject); | Port | Image14 | Открытие картинки в большой форме |
| procedure TPortf.Image14MouseEnter(Sender: TObject); | Port | Image14 | Появление надписи «Press me» |
| procedure TPortf.Image14MouseLeave(Sender: TObject); | Port | Image14 | Исчезновение надписи «Press me» |
| procedure TPortf.Image15Click(Sender: TObject); | Port | Image | Кнопка «Назад» |
| procedure TTestForm.Button1Click(Sender: TObject); | Test | Button1 | Тест |
| procedure TDPict.Image3Click(Sender: TObject); | dpictur | Image3 | Кнопка «Назад» |
| procedure TChoiseTestForm.BitBtn1Click(Sender: TObject); | Choisetest | BitBtn1 | Выбор один из вариантов теста |
| procedure TChoiseTestForm.BitBtn2Click(Sender: TObject); | ChoiseTest | BitBtn | Закрыть тест |
| procedure TVideo.Image3Click(Sender: TObject); | Video | Button1 | Вкл. видео |
| procedure TVideo.FormCreate(Sender: TObject); | Video | Form | Загрузка видео |
| procedure TVideo.Image15Click(Sender: TObject); | Video | Image15 | Переход на предыдущую форму |

**3.1.3 Описание использованных компонентов**

Описание использованных при разработке приложения компонентов приводиться в таблице 2

Таблица 2 – Использованные компоненты

|  |  |  |
| --- | --- | --- |
| Компонент | На какой форме расположено | Назначение |
| TMaimMenu | Avt,choiseTest,dpictur,main,  Port,test,ther,vide | Используется для создания главного меню проекта |
| TImage | Avt,choiseTest,dpictur,main,  Port,test,ther,vide | Картинки на формах |
| TWebBrowser1 | Theory | Браузер для чтения |
| TButton1 | TestForm | Кнопка для продолжения теста |
| TBitBtn1 | ChoiseTestForm | Продалжить |
| TBitBtn2 | ChoiseTestForm | Закрыть тест |
| TLabel1 | TestForm | Вывод информации |
| TRadioGroup1 | TestForm | Выбор из нескольких вариантов ответа |
| TRadioButton1 | ChoiseTestForm | Выбор варианта теста |
| TRadioButton2 | ChoiseTestForm | Выбор варианта теста |

**3.2 Спецификация программы**

Точное название проекта и его состав приводится в таблице 3.

Таблица 3 – Спецификация программы

| Имя файла | Назначение |
| --- | --- |
| Blender\_Pr.exe | Исполняемый файл проекта, используется для запуска программы на выполнение. |
| Blender\_Pr.dproj | Файл проекта, связывает все файлы из которых состоит приложение. |
| Main.pas | Файл программного модуля главной формы |
| Ther.pas | Файл программного модуля теоретической части проекта |
| Port.pas | Файл программного модуля содержащий портфолио |
| Test.pas | Файл программного модуля для прохождения теста |
| ChoiseTest.pas | Файл программного модуля для выбора варианта теста |
| Dpictur.pas | Файл программного модуля с изображением большего размера |
| Avt.pas | Файл программного модуля с информацией об авторе |
| Vide.pas | Файл программного модуля с видеоизображением |
| Main.dfm | Главная форма |
| Ther.dfm | Форма с теоретическим материалом |
| Port.dfm | Форма с авторскими работами |
| Test.dfm | Форма с тестом |
| ChoiseTest.dfm | Форма с выборам одного из теста |
| Dpictur.dfm | Форма с увеличенным изображением |
| Avt.dfm | Форма об Авторе |

Продолжение таблицы 3

|  |  |
| --- | --- |
| Имя файла | Назначение |
| Vide.dfm | Форма с видео |
| Spravka.chm | Справочная информация |
| COMP.txt | Тест про компоненты blender |
| HBLD.txt | Тест про историю blender |
| Setup.exe | Установочный пакет программы |
| Theory.htm | Теоретический материал |

**4 Тестирование**

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

Отчет о результатах тестирования представлены в таблице 4

Таблица 4 - Отчет о результатах тестирования.

|  |  |  |  |
| --- | --- | --- | --- |
| Тест | Ожидаемый результат | Фактический результат | Результат тестирования |
| Проверка пункта меню «Пройти тест» | Переход на форму выбора теста | Переход на форму | Выполнено |
| Проверка пункта меню «Выход» | Выход из проекта | Выход из проекта | Выполнено |
| Проверка пункта меню «Об авторе» | Открытие формы «об авторе» | Открылась форма «Об авторе» | Выполнено |
| Проверка пункта меню «На главную» | Открытие главной формы | Открылась главной форме | Выполнено |
| Проверка кнопки меню «справка» | Открытие справочной инф. | Открылась справочная инф. | Выполнено |
| Проверка пункта меню «Справка» | Открытие формы «Справка» | Открылась форма «Справка» | Выполнено |
| Проверка кнопки «Теория» | Открытие формы «Теория» | Открылась форма «Теория» | Выполнено |
| Проверка кнопки «Портфолио» | Открытие формы «Портфолио» | Открылась форма «Портфолио» | Выполнено |
| Проверка кнопки «назад» | Переход на предыдущую форму | Переход на предыдущую форму | Выполнено |
| Проверка двойного нажатия на картинку | Открытие формы с увеличенным изображение | Открылась форма с увеличенным изображение | Выполнено |
| Проверка кнопок выбора теста | Открытие определенного из тестов | Открылся определенный тест | Выполнено |
| Проверка кнопки «<<<» | Переход на предыдущую форму | Переход на предыдущую форму | Выполнено |
| Проверка кнопки «>>>» | Переход на cледующую форму | Переход на селудющую форму | Выполнено |

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

При использовании данного электронного средства обучения пользователь не столкнуться с ошибками системы

**5 Руководство пользователя**

**5.1 Общие сведения о программном продукте**

Разработанная программа является электронным средством обучения «Проведение фотосессии».

Данный средство обучения предназначено для обучения основам проведения фотосессии и настройке камеры.

Основными функциями приложения являются:

- Изучение теоретической информации по проведению фотосессии, настройки камеры;

- Просмотр видеоизображения про то , как настроить идеально фотоаппарат;

- Прохождение теста «Проведение фотосессии+компоненты фотоаппарата»

- Изучение справочной информации про функционал программы

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

Тестирование проводилось на различных классах ЭВМ и работать с данной программой было комфортно. Но даже на системе следующего типа работать было комфортно:

Разработка происходила на ПК со следующими характеристиками:

- Процессор : Intel(R) Pentium(R) Silver N5000;

- ОЗУ: 4Gb;

- Память: SSD 1028Gb;

- ОС – Windows 10.

**5.2 Инсталляция**

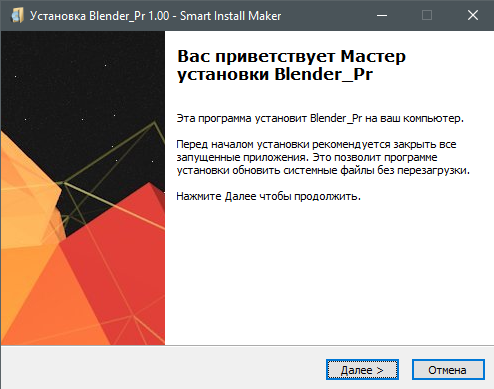
Для того что бы установить программу необходимо запустить файл Blender\_Project.exe. Появится окно установки обучающего приложения Blender\_Project (Рисунок 8).

Рисунок 8 – Установка обучающего приложения Blender\_Pr

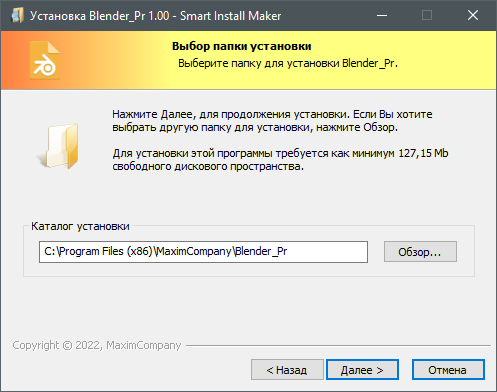
После нажатия на кнопку далее появится окно выбора каталога установки (Рисунок 9).

Рисунок 9 –Выбор каталога установки

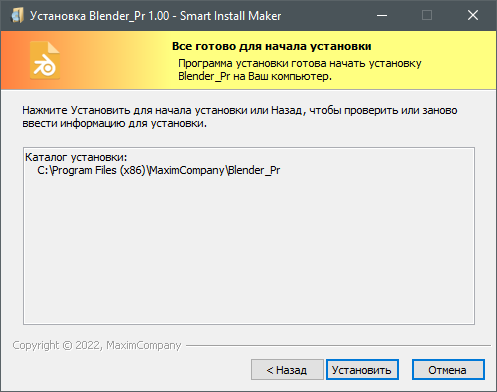
После выбора каталога, нажимаем «Далее», после нам показывают каталог установки. Если же все устраивает, нажимаем «Установить» (Рисунок 10).

Рисунок 10 – выбранный каталог установки

Затем происходит установки Программы, следует только ждать (Рисунок 11):

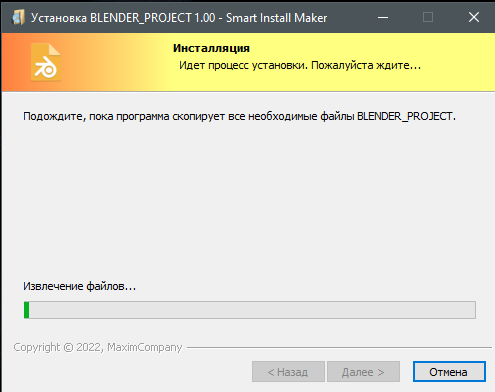


Рисунок 11 – Установка файлов

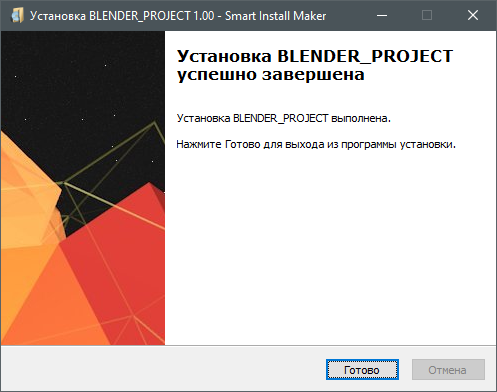
После установки, можно найти каталог с скачанными файлами (Рисунок 12):

Рисунок 12 – Файлы установлены

**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 клавишей мыши на ярлык под названием Blender\_Project.exe (Рисунок 13):



Рисунок 13 – Ярлык программы

Вторым способом является запуск из каталога, в который установилась программа.

**5.3.2 Инструкции по работе с программой**

После запуска программы на экране появляется главный экран (Рисунок 14).

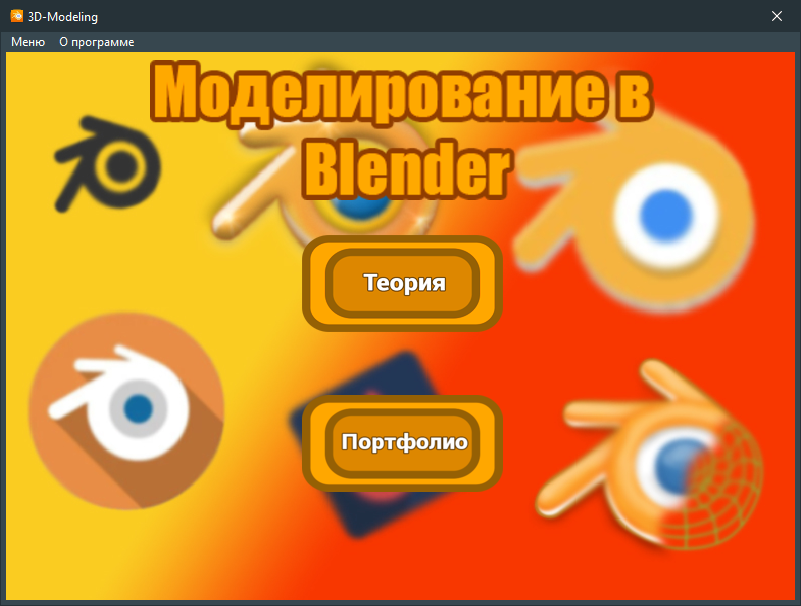


Рисунок 14 – Главный экран

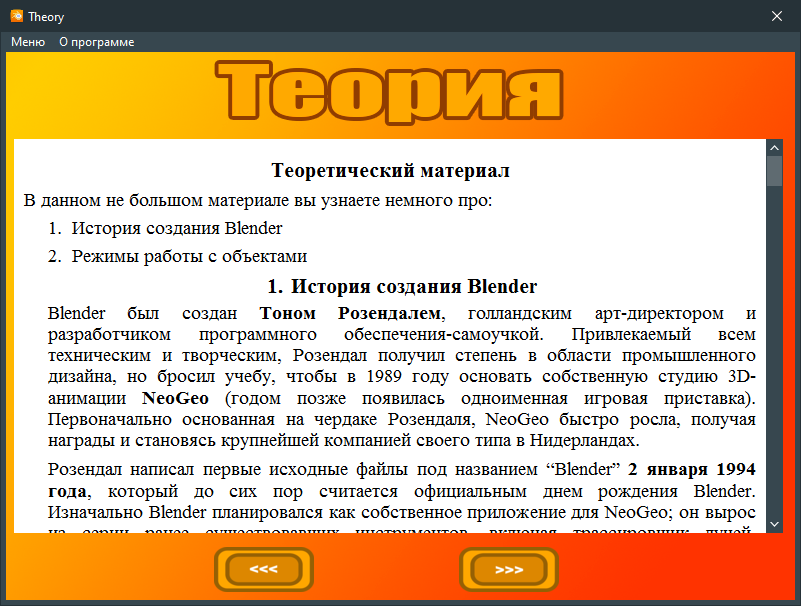
После нажатия на кнопку «Теория», появиться окно с теоретическим материалом (Рисунок 15).

Рисунок 15 – Окно с теоретическим материалом

Затем прочитав материал можно перейти на окно с видео, нажав на кнопку «>>>» (Рисунок 16).

Рисунок 16 – Окно с видео

Прочитав информацию и просмотрев видео, можно перейти на окно с авторскими работами (Рисунок 17).

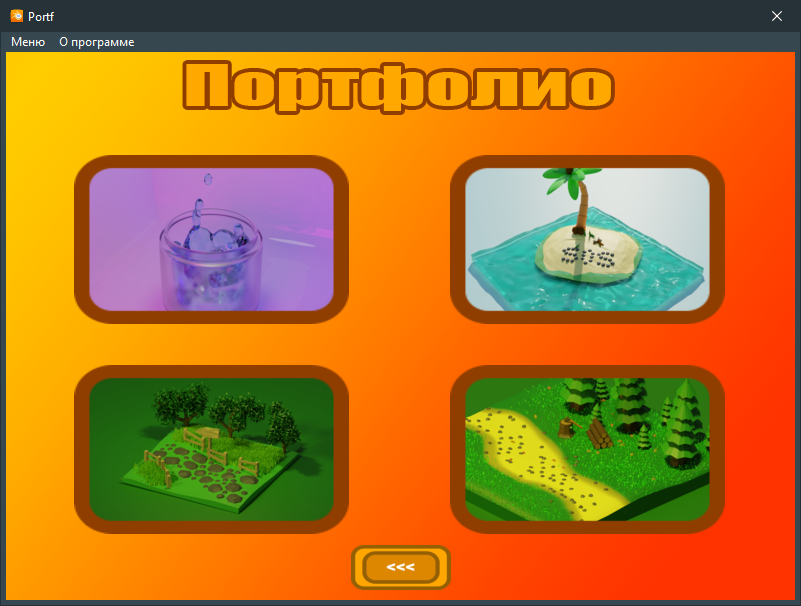


Рисунок 17 – Окно с работами

Двойным нажатием на картинку можно открыть ее в увеличенном масштабе (Рисунок 18).

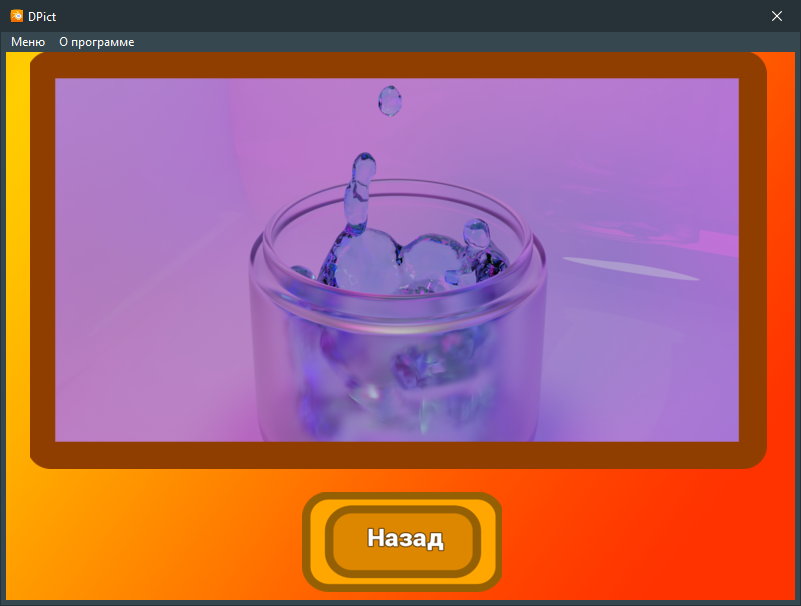


Рисунок 18 – Увеличенное изображение

После всех действий можно пройти тестовую часть проекта (Рисунок 19).

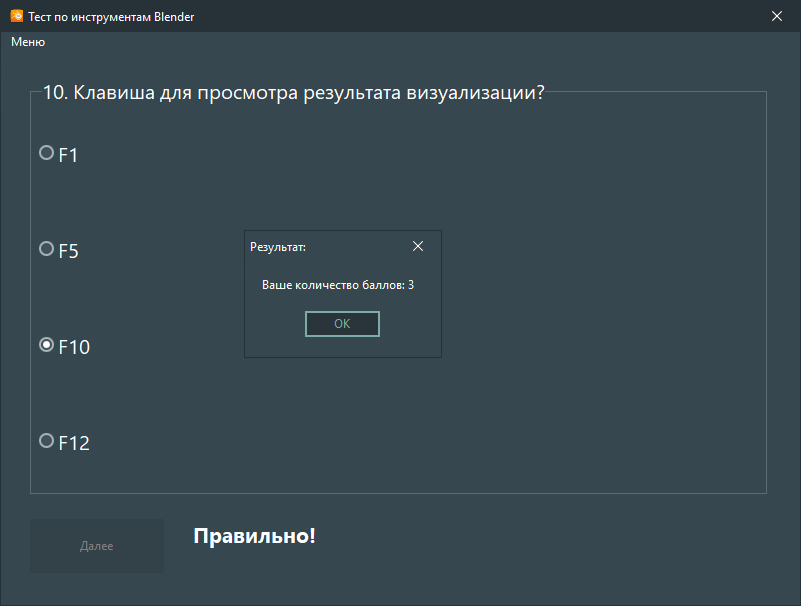
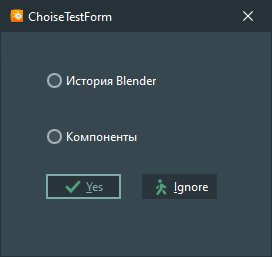


Рисунок 19 – Тестовая часть

Перейди по кнопке «Об авторе» можно прочитать информацию об авторе (Рисунок 20).

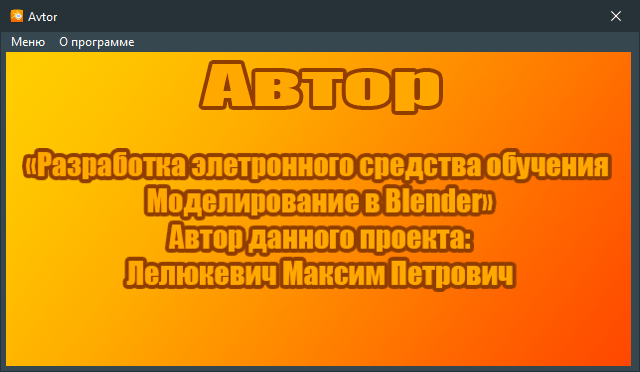


Рисунок 20 – Информационное окно об авторе

**5.3.3 Завершение работы с программой**

Завершить работу с программой можно единственным способом через Меню - Выход (Рисунок 21).

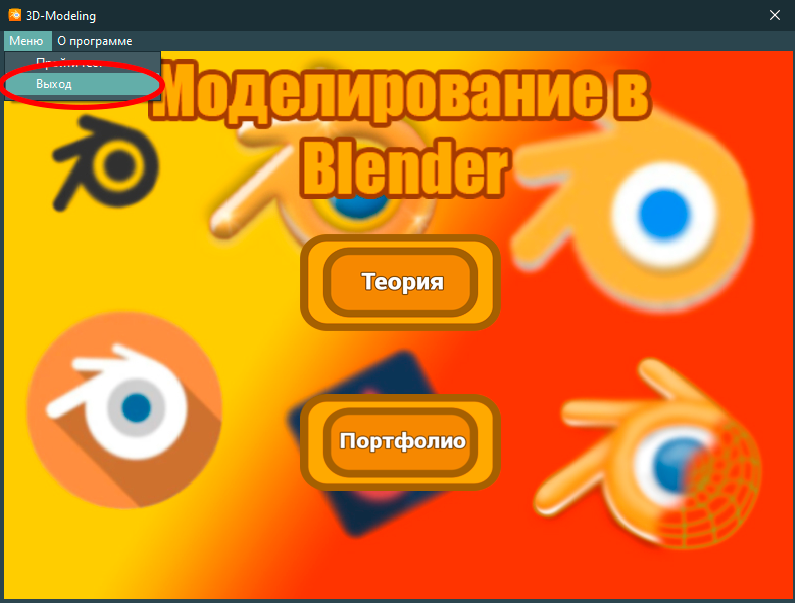


Рисунок 21 – Выход из программы

**5.4 Использование системы справочной информации**

Справочную систему можно запустить с помощью команды меню О программе - Справка (Рисунок 22).

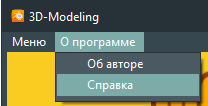


Рисунок 22 – справка

После нажатия появиться окно с справочной информацией о программе (Рисунок 23).

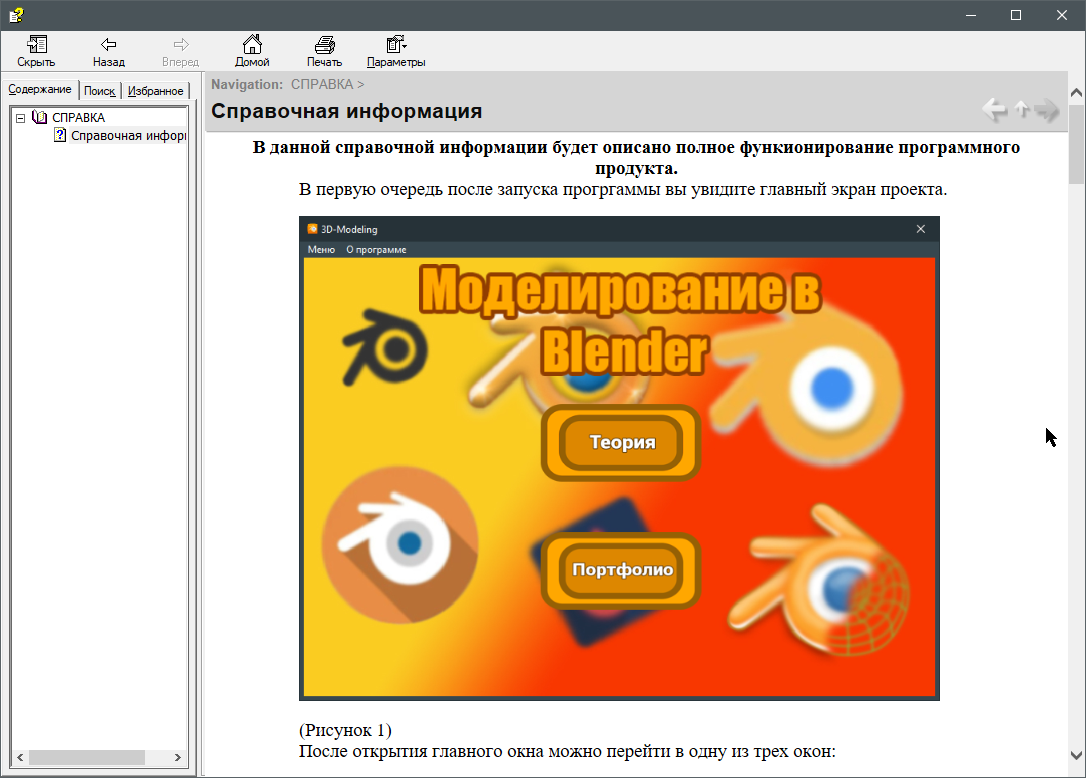


Рисунок 23 – Справочная информация

**Заключение**

Цель данного проекта была реализована - создано электронное средство обучения «Моделирование в Blender».

Данное электронное средство обучения будет интересно людям любого возраста, желающим познакомиться с моделированием в графическом редакторе Blender.

Был реализован простой и понятный пользовательский интерфейс, который позволяет использовать приложение пользователю, не обладающему дополнительными знаниями ЭВМ.

В ходе реализации поставленной задачи был укреплён пройденный курс программирования в среде Delphi, а также получено много дополнительной информации о её возможностях.

После тщательного тестирования приложения были выявлены некоторые недоработки, которые были частично исправлены на стадии проектирования, и полностью исключены на стадии тестирования программы. В целом при реализации программы, были выполнены все условия, перечисленные в предыдущих разделах пояснительной записки.

Таким образом, можно сказать, что поставленная задача была реализована успешно.

**Список используемых источников**

1. Уроки Delphi. [Видео ресурс]. – Режим доступа: <https://www.youtube.com/watch?v=y6WUUXWMk1E>. – Дата доступа 18.06.2022.

2. Форум профессиональных и начинающих программистов. [Электронный ресурс]. – Режим доступа: <https://www.programmersforum.ru/>. – Дата доступа 20.06.2022.

3. Иллюстрированный самоучитель по Delphi [Электронный ресурс]. – Режим доступа: [samoychiteli.ru/document27136.htm](file:///C:\\Users\\User\\Downloads\\samoychiteli.ru\\document27136.htm). – Дата доступа 22.06.2022.

4. Электронная книга по Delphi [Электронный ресурс]. – Режим доступа: <https://lp.embarcadero.com/DelphiBookInRussia>. – Режим доступа 24.06.2022.

5. Иллюстрации для проекта – Режим доступа: <https://icon-icons.com/ru/Поиск/иконки/?filtro=Blender&sort=popular>. – Режим доступа 25.06.2022.

6. Исторические материалы про Blender – Режим доступа: <https://www.blender.org/about/history/>. – Режим доступа 25.06.2022.

Приложение A

Листинг программы

unit Main; (**Главная форма**)

interface

uses

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

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

Vcl.ExtDlgs, Vcl.Imaging.jpeg, Vcl.Imaging.pngimage,ShellAPI ;

type

TMainForm = class(TForm)

MainMenu1: TMainMenu;

N1: TMenuItem;

N3: TMenuItem;

N4: TMenuItem;

N5: TMenuItem;

N8: TMenuItem;

N9: TMenuItem;

Image1: TImage;

Image4: TImage;

Image3: TImage;

Image5: TImage;

procedure N8Click(Sender: TObject);

procedure N9Click(Sender: TObject);

procedure Image4Click(Sender: TObject);

procedure Image3Click(Sender: TObject);

procedure N4Click(Sender: TObject);

procedure N5Click(Sender: TObject);

private

{ Private declarations }

public

{ Public declarations }

end;

var

MainForm: TMainForm;

implementation

{$R \*.dfm}

uses ChoiseTest, Test, Ther, Port, avt;

procedure TMainForm.Image3Click(Sender: TObject);

begin

MainForm.Hide;

Theory.Show;

Theory.WebBrowser1.Navigate(ExtractFilePath(ParamStr(0)) + 'Docs\Theory.htm')

end;

procedure TMainForm.Image4Click(Sender: TObject);

begin

MainForm.Hide;

Portf.Show;

end;

procedure TMainForm.N4Click(Sender: TObject);

begin

MainForm.Hide;

Avtor.Show;

end;

procedure TMainForm.N5Click(Sender: TObject);

begin

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

end;

procedure TMainForm.N8Click(Sender: TObject);

begin

MainForm.Close;

end;

procedure TMainForm.N9Click(Sender: TObject);

begin

ChoiseTestForm.ShowModal;

end;

end.

unit Port; ( **Портфолио**)

interface

uses

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

Vcl.Controls, Vcl.Forms, Vcl.Dialogs, Vcl.Menus, Vcl.Imaging.jpeg,

Vcl.ExtCtrls, Vcl.OleCtrls, SHDocVw, Vcl.Imaging.pngimage, Vcl.StdCtrls,ShellAPI ;

type

TPortf = class(TForm)

Image1: TImage;

MainMenu1: TMainMenu;

N1: TMenuItem;

N7: TMenuItem;

N2: TMenuItem;

N3: TMenuItem;

N4: TMenuItem;

N5: TMenuItem;

N6: TMenuItem;

Image2: TImage;

Image3: TImage;

Image4: TImage;

Image5: TImage;

Image6: TImage;

Image7: TImage;

Image8: TImage;

Image9: TImage;

Image10: TImage;

Image11: TImage;

Image12: TImage;

Image13: TImage;

Image14: TImage;

Image15: TImage;

procedure N2Click(Sender: TObject);

procedure N3Click(Sender: TObject);

procedure N7Click(Sender: TObject);

procedure Image11MouseEnter(Sender: TObject);

procedure Image11MouseLeave(Sender: TObject);

procedure Image14MouseEnter(Sender: TObject);

procedure Image14MouseLeave(Sender: TObject);

procedure Image12MouseEnter(Sender: TObject);

procedure Image12MouseLeave(Sender: TObject);

procedure Image13MouseEnter(Sender: TObject);

procedure Image13MouseLeave(Sender: TObject);

procedure Image14DblClick(Sender: TObject);

procedure Image12DblClick(Sender: TObject);

procedure Image13DblClick(Sender: TObject);

procedure Image11DblClick(Sender: TObject);

procedure Image15Click(Sender: TObject);

procedure N6Click(Sender: TObject);

procedure N5Click(Sender: TObject);

private

{ Private declarations }

public

{ Public declarations }

end;

var

Portf: TPortf;

implementation

{$R \*.dfm}

uses Main, ChoiseTest, dpictur, avt;

procedure TPortf.Image11DblClick(Sender: TObject);

begin

Portf.Hide;

DPict.Show;

Dpict.Image2.Picture.LoadFromFile(ExtractFilePath(ParamStr(0)) + 'cup.png')

end;

procedure TPortf.Image11MouseEnter(Sender: TObject);

begin

Image7.Visible:=True;

end;

procedure TPortf.Image11MouseLeave(Sender: TObject);

begin

Image7.Visible:=False;

end;

procedure TPortf.Image12DblClick(Sender: TObject);

begin

Portf.Hide;

DPict.Show;

Dpict.Image2.Picture.LoadFromFile(ExtractFilePath(ParamStr(0)) + 'forest.png')

end;

procedure TPortf.Image12MouseEnter(Sender: TObject);

begin

Image10.Visible:=True;

end;

procedure TPortf.Image12MouseLeave(Sender: TObject);

begin

Image10.Visible:=False;

end;

procedure TPortf.Image13DblClick(Sender: TObject);

begin

Portf.Hide;

DPict.Show;

Dpict.Image2.Picture.LoadFromFile(ExtractFilePath(ParamStr(0)) + 'garden.png')

end;

procedure TPortf.Image13MouseEnter(Sender: TObject);

begin

Image9.Visible:=True;

end;

procedure TPortf.Image13MouseLeave(Sender: TObject);

begin

Image9.Visible:=False;

end;

procedure TPortf.Image14DblClick(Sender: TObject);

begin

Portf.Hide;

DPict.Show;

Dpict.Image2.Picture.LoadFromFile(ExtractFilePath(ParamStr(0)) + 'island.png');

end;

procedure TPortf.Image14MouseEnter(Sender: TObject);

begin

Image8.Visible:=True;

end;

procedure TPortf.Image14MouseLeave(Sender: TObject);

begin

Image8.Visible:=False;

end;

procedure TPortf.Image15Click(Sender: TObject);

begin

Portf.Hide;

MainForm.Show;

end;

procedure TPortf.N2Click(Sender: TObject);

begin

ChoiseTestForm.ShowModal;

end;

procedure TPortf.N3Click(Sender: TObject);

begin

MainForm.Close;

end;

procedure TPortf.N5Click(Sender: TObject);

begin

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

end;

procedure TPortf.N6Click(Sender: TObject);

begin

Portf.Hide;

Avtor.Show;

end;

procedure TPortf.N7Click(Sender: TObject);

begin

Portf.Hide;

MainForm.Show;

end;

end.

unit Ther; (**Теоритический материал**)

interface

uses

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

Vcl.Controls, Vcl.Forms, Vcl.Dialogs, Vcl.Imaging.jpeg, Vcl.ExtCtrls,

Vcl.Menus, Vcl.Imaging.pngimage, Vcl.StdCtrls, Vcl.OleCtrls, SHDocVw,ShellAPI ;

type

TTheory = class(TForm)

Image1: TImage;

MainMenu1: TMainMenu;

N1: TMenuItem;

N9: TMenuItem;

N8: TMenuItem;

N3: TMenuItem;

N4: TMenuItem;

N5: TMenuItem;

N2: TMenuItem;

Image2: TImage;

WebBrowser1: TWebBrowser;

Image15: TImage;

Image3: TImage;

procedure N2Click(Sender: TObject);

procedure N8Click(Sender: TObject);

procedure N9Click(Sender: TObject);

procedure Image3Click(Sender: TObject);

procedure Image15Click(Sender: TObject);

procedure N4Click(Sender: TObject);

procedure N5Click(Sender: TObject);

private

{ Private declarations }

public

{ Public declarations }

end;

var

Theory: TTheory;

implementation

{$R \*.dfm}

uses Main, ChoiseTest, Vide, avt;

procedure TTheory.Image15Click(Sender: TObject);

begin

Theory.Hide;

MainForm.Show;

end;

procedure TTheory.Image3Click(Sender: TObject);

begin

Theory.Hide;

Video.Show;

begin

Video.MediaPlayer1.Open

end;

Video.MediaPlayer1.Play

end;

procedure TTheory.N2Click(Sender: TObject);

begin

Theory.Hide;

MainForm.Show;

end;

procedure TTheory.N4Click(Sender: TObject);

begin

Theory.Hide;

Avtor.Show;

end;

procedure TTheory.N5Click(Sender: TObject);

begin

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

end;

procedure TTheory.N8Click(Sender: TObject);

begin

MainForm.Close;

end;

procedure TTheory.N9Click(Sender: TObject);

begin

ChoiseTestForm.ShowModal;

end;

end.

unit Test; (**Тест**)

interface

uses

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

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

type

TTestForm = class(TForm)

RadioGroup1: TRadioGroup;

Button1: TButton;

Label1: TLabel;

MainMenu1: TMainMenu;

N1: TMenuItem;

N2: TMenuItem;

N3: TMenuItem;

N4: TMenuItem;

procedure Button1Click(Sender: TObject);

procedure N2Click(Sender: TObject);

procedure N3Click(Sender: TObject);

procedure N4Click(Sender: TObject);

private

{ Private declarations }

public

{ Public declarations }

end;

var

TestForm: TTestForm;

implementation

{$R \*.dfm}

uses Main, ChoiseTest;

procedure TTestForm.Button1Click(Sender: TObject);

begin

if (RadioGroup1.ItemIndex>-1) and (not Eof(f)) then

begin

if RadioGroup1.ItemIndex = Nvern-1 then

begin

Label1.Caption := 'Правильно!';

ball:=ball+1; //Если выбранный вариант соответствует номеру верного ответа, то балл прибавляется и очищается поле для следующего вопроса

end

else

Label1.Caption := 'Не правильно!';

RadioGroup1.Items.Clear;

Repeat

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

begin

delete(s,1,1);

RadioGroup1.Caption:=s;

end

else

if s[1]='\*' then

begin

delete(s,1,1);

Nvern:=StrToInt(s);

end

else

RadioGroup1.Items.Add(s);

readln(f,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 RadioGroup1.ItemIndex = Nvern-1 then

ball:=ball+1;

CloseFile(f);

Button1.Enabled := False;

Application.Title := 'Результат:';

if MessageDlg('Ваше количество баллов: ' + IntToStr(ball), mtCustom, [mbOK], 0) = mrOK then

MainForm.Show;

TestForm.Hide;

end;

end;

procedure TTestForm.N4Click(Sender: TObject);

begin

MainForm.Close;

end;

procedure TTestForm.N3Click(Sender: TObject);

begin

TestForm.Hide;

MainForm.Show;

end;

procedure TTestForm.N2Click(Sender: TObject);

begin

ChoiseTestForm.Showmodal;

end;

end.

unit Vide; (**Видеоматериал**)

interface

uses

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

Vcl.Controls, Vcl.Forms, Vcl.Dialogs, Vcl.Imaging.pngimage, Vcl.ExtCtrls,

Vcl.MPlayer, Vcl.StdCtrls, Vcl.Menus,ShellAPI;

type

TVideo = class(TForm)

Image1: TImage;

Image2: TImage;

MediaPlayer1: TMediaPlayer;

MainMenu1: TMainMenu;

N1: TMenuItem;

N9: TMenuItem;

N8: TMenuItem;

N3: TMenuItem;

N4: TMenuItem;

N5: TMenuItem;

N2: TMenuItem;

Image15: TImage;

Panel1: TPanel;

procedure FormCreate(Sender: TObject);

procedure Button1Click(Sender: TObject);

procedure N2Click(Sender: TObject);

procedure Image15Click(Sender: TObject);

procedure N4Click(Sender: TObject);

procedure N8Click(Sender: TObject);

procedure N9Click(Sender: TObject);

procedure N5Click(Sender: TObject);

private

{ Private declarations }

public

{ Public declarations }

end;

var

Video: TVideo;

implementation

{$R \*.dfm}

uses Ther, Main, avt, ChoiseTest;

procedure TVideo.FormCreate(Sender: TObject);

begin

MediaPlayer1.FileName:=(ExtractFilePath(ParamStr(0)) + 'cupp.wmv');

end;

procedure TVideo.Image15Click(Sender: TObject);

begin

Video.Hide;

Theory.Show;

MediaPlayer1.Stop;

MediaPlayer1.Close;

end;

procedure TVideo.N2Click(Sender: TObject);

begin

Video.Hide;

MainForm.Show;

end;

procedure TVideo.N4Click(Sender: TObject);

begin

Video.Hide;

Avtor.Show;

end;

procedure TVideo.N5Click(Sender: TObject);

begin

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

end;

procedure TVideo.N8Click(Sender: TObject);

begin

MainForm.Close;

end;

procedure TVideo.N9Click(Sender: TObject);

begin

ChoiseTestForm.ShowModal;

end;

procedure TVideo.Button1Click(Sender: TObject);

Begin

MediaPlayer1.Open;

MediaPlayer1.Play;

End;

end.

unit dpictur; (**Увеличенное изображение**)

interface

uses

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

Vcl.Controls, Vcl.Forms, Vcl.Dialogs, Vcl.Imaging.jpeg, Vcl.ExtCtrls,

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

type

TDPict = class(TForm)

Image1: TImage;

MainMenu1: TMainMenu;

N1: TMenuItem;

N7: TMenuItem;

N2: TMenuItem;

N3: TMenuItem;

N4: TMenuItem;

N5: TMenuItem;

N6: TMenuItem;

Image2: TImage;

Image3: TImage;

Image4: TImage;

procedure Image3Click(Sender: TObject);

procedure N6Click(Sender: TObject);

procedure N3Click(Sender: TObject);

procedure N2Click(Sender: TObject);

procedure N5Click(Sender: TObject);

procedure N7Click(Sender: TObject);

private

{ Private declarations }

public

{ Public declarations }

end;

var

DPict: TDPict;

implementation

{$R \*.dfm}

uses Port, avt, ChoiseTest, Main;

procedure TDPict.Image3Click(Sender: TObject);

begin

Dpict.Hide;

Portf.Show;

end;

procedure TDPict.N2Click(Sender: TObject);

begin

ChoiseTestForm.ShowModal;

end;

procedure TDPict.N3Click(Sender: TObject);

begin

MainForm.close;

end;

procedure TDPict.N5Click(Sender: TObject);

begin

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

end;

procedure TDPict.N6Click(Sender: TObject);

begin

Dpict.Hide;

Avtor.Show;

end;

procedure TDPict.N7Click(Sender: TObject);

begin

Dpict.Hide;

mainForm.Show;

end;

end.

unit ChoiseTest; (**Выбор теста**)

interface

uses

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

Vcl.Controls, Vcl.Forms, Vcl.Dialogs, Vcl.StdCtrls, Vcl.Buttons;

type

TChoiseTestForm = class(TForm)

RadioButton1: TRadioButton;

RadioButton2: TRadioButton;

BitBtn1: TBitBtn;

BitBtn2: TBitBtn;

procedure BitBtn2Click(Sender: TObject);

procedure BitBtn1Click(Sender: TObject);

private

{ Private declarations }

public

{ Public declarations }

end;

var

ChoiseTestForm: TChoiseTestForm;

f:text;

s:string;

ball:integer;

Nvern:integer;

implementation

{$R \*.dfm}

uses Test, Main;

procedure TChoiseTestForm.BitBtn1Click(Sender: TObject);

begin

if RadioButton1.Checked then

begin

MainForm.Hide;

TestForm.Free;

Application.CreateForm(TTestForm, TestForm);

TestForm.Show;

ChoiseTestForm.Close;

TestForm.Caption := 'Тест по Истории Blender';

TestForm.Button1.Enabled:=True;

AssignFile(f ,ExtractFilePath(ParamStr(0)) + 'HBLD.txt');

reset(f); //Открываем файл для чтения

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

ball:=0;

repeat

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

begin //Если первый символ строки ‘-‘ значит это вопрос

delete(s,1,1);

TestForm.RadioGroup1.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

TestForm.RadioGroup1.Items.Add(s); //Иначе это вариант ответа

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

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

end;

if RadioButton2.Checked then

begin

MainForm.Hide;

TestForm.Free;

Application.CreateForm(TTestForm, TestForm);

TestForm.Show;

ChoiseTestForm.close;

TestForm.Caption := 'Тест по инструментам Blender';

TestForm.Button1.Enabled:=True;

AssignFile(f ,ExtractFilePath(ParamStr(0)) + 'COMP.txt');

reset(f); //Открываем файл для чтения

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

ball:=0;

repeat

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

begin //Если первый символ строки ‘-‘ значит это вопрос

delete(s,1,1);

TestForm.RadioGroup1.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

TestForm.RadioGroup1.Items.Add(s); //Иначе это вариант ответа

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

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

end;

end;

procedure TChoiseTestForm.BitBtn2Click(Sender: TObject);

begin

close

end;

end.

unit avt; (**Об авторе**)

interface

uses

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

Vcl.Controls, Vcl.Forms, Vcl.Dialogs, Vcl.Imaging.pngimage, Vcl.ExtCtrls,

Vcl.Menus, Vcl.StdCtrls,ShellAPI ;

type

TAvtor = class(TForm)

Image1: TImage;

Image2: TImage;

MainMenu1: TMainMenu;

N1: TMenuItem;

N2: TMenuItem;

N3: TMenuItem;

N4: TMenuItem;

N5: TMenuItem;

Image3: TImage;

Image4: TImage;

procedure N3Click(Sender: TObject);

procedure N2Click(Sender: TObject);

procedure Image4Click(Sender: TObject);

procedure N5Click(Sender: TObject);

private

{ Private declarations }

public

{ Public declarations }

end;

var

Avtor: TAvtor;

implementation

{$R \*.dfm}

uses Port, Main;

procedure TAvtor.Image4Click(Sender: TObject);

begin

ShowMessage('Ты нашел пасхалку:)')

end;

procedure TAvtor.N2Click(Sender: TObject);

begin

Avtor.hide;

MainForm.Show;

end;

procedure TAvtor.N3Click(Sender: TObject);

begin

MainForm.Close;

end;

procedure TAvtor.N5Click(Sender: TObject);

begin

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

end;

end.