**Введение**

На курсовое проектирование была поставлена задача — разработать игровое приложение на тему: «Автобазар».

Целью курсового проекта является создание интерактивного игрового приложения, в котором пользователь может в простой и увлекательной форме взаимодействовать с игровым процессом, основанным на выборе автомобилей. Игровое приложение «Автобазар» предназначено для развития финансовой грамотности, сообразительности и внимательности у игроков, а также демонстрирует навыки работы с графическим интерфейсом и логикой игровых механик, реализованных на платформе C++ CLI с использованием Windows Forms.

Создаваемое приложение будет рассчитано на пользователей самого различного уровня подготовки. Применение игрового приложения «Автобазар» возможно как в учебных целях — для демонстрации основ создания Windows-приложений, так и просто для проведения досуга.

Игровое приложение состоит из трёх основных форм: экрана загрузки, главного меню и игровой области. В меню игрок может начать игру, ознакомиться с правилами, справкой, информацией об авторе. В самой игре реализована механика выбора автомобиля. За правильный выбор игрок получает игровую валюту, за неправильный — теряет. При достижении конечного результата игра завершается с сообщением о вашем уровне знаний в области покупки автомобилей.

Далее приведём краткое описание разделов пояснительной записки.

Первый раздел носит название «Анализ задачи». В нём вы сможете ознакомиться с постановкой задачи, которая включает в себя: исследование предметной области автомобильных аркадных игр, описание пользовательских требований и логики поведения приложения. Также в этом разделе описываются существующие аналоги и методы реализации подобных задач. Приведены все входные и выходные данные, необходимые для функционирования приложения. В подразделе «Инструменты разработки» рассмотрена программная среда, в которой создаётся данный проект. Также приведены минимальные и оптимальные аппаратные характеристики, обеспечивающие корректную работу игрового приложения.

В разделе «Проектирование задачи» будут рассмотрены основные аспекты построения игрового продукта. Здесь будет подробно описана структура пользовательского интерфейса, способы навигации между формами, логика обработки игровых событий и реализация алгоритмов начисления и списания игровой валюты. Кроме того, будет описана система помощи и настройки.

Раздел «Реализация задачи» посвящён описанию конкретной реализации элементов интерфейса и игровых компонентов. Здесь будет дано описание всех функций, кнопок, игровых объектов и файлов, используемых в проекте.

Четвёртый раздел — «Тестирование». В нём будет подробно описано тестирование всех функций приложения. Будет оттестирован каждый элемент меню и игровой логики. Также будут рассмотрены различные сценарии взаимодействия с приложением, от обычного игрового сеанса до обработки ошибок и граничных ситуаций (например, проигрыш).

В разделе «Применение» будет описано назначение, область применения и среда функционирования игрового приложения «Автобазар». Также будет раскрыт вопрос о возможности использования приложения в демонстрационных, учебных или развлекательных целях.

«Заключение» будет содержать краткую формулировку поставленной задачи, основные итоги и результаты проделанной работы, список использованных средств и технологий. Также будет оценена степень реализации задач автоматизации на различных этапах разработки.

В разделе «Литература» будет приведён список использованных источников, включая материалы по программированию на C++/CLI, документацию по Windows Forms, а также статьи и ресурсы, связанные с игровыми алгоритмами и обработкой пользовательского ввода.

В Приложениях к пояснительной записке будет приведён листинг исходного кода с необходимыми пояснениями и комментариями.

Графическая часть пояснительной записки будет включать схему архитектуры приложения и визуальное представление интерфейса, демонстрирующее взаимодействие между формами и компонентами.

**1 Анализ задачи**

* 1. **Постановка задачи**

Игровая индустрия является одним из наиболее активно развивающихся направлений в сфере информационных технологий. Современные игры выполняют не только развлекательную функцию, но также развивают реакцию, внимание, точность и координацию движений. Особенно востребованными являются простые аркадные игры, позволяющие пользователю проводить короткие игровые сессии без необходимости глубоко погружаться в сюжет или сложную механику. Одной из таких классических аркадных игр является «Автобазар» — игровое приложение, в котором пользователю предлагается выбирать автомобиль, набирая очки и соревнуясь с самим собой.

В настоящее время аналогичные задачи решаются с использованием как браузерных, так и мобильных платформ. Тем не менее, существуют потребности в реализации подобных решений и на настольных системах Windows, особенно в рамках учебных, демонстрационных и любительских проектов. Разработка игрового приложения «Автобазар» в среде Windows Forms на языке C++/CLI позволяет продемонстрировать основные навыки программирования интерфейсов, обработки событий, использования графики и логики игрового процесса в настольной ОС.

Периодичность использования данной программы не ограничена — она может быть запущена пользователем в любое время по желанию. Основная цель приложения — обеспечить простую и доступную игровую среду, где пользователь может отдохнуть, развить свои знания в области продажи автомобилей. Разрабатываемое программное средство упрощает реализацию и изучение подобных проектов за счёт доступной архитектуры, модульности, интуитивного интерфейса и понятной логики.

Существующие аналоги представлены в виде классических аркад (Car For Sale Simulator, Симулятор перекупа 2023, браузерные игры), однако они, как правило, не разрабатываются с использованием Windows Forms и не обладают открытым исходным кодом, что затрудняет их использование в учебных целях. Данное приложение создается как легко масштабируемый и расширяемый проект, адаптируемый под новые функции и улучшения.

Выходные данные отображаются в форме визуальных элементов графического интерфейса:

* текущий счёт (выводится в виде текстового поля);
* надпись вашего рейтинга при достижении минимального или максимального счета;

Все данные представлены в интерактивной экранной форме, которая обновляется в реальном времени с помощью таймеров и событий.

Ввод данных осуществляется посредством взаимодействия пользователя с мышью и элементами управления:

* щелчки мыши по кнопкам;
* нажатия на кнопки меню: запуск игры, выход, справка и т. д.;
* включение/выключение звука;

Структура входных данных не требует предварительной подготовки — данные поступают напрямую от действий пользователя, обрабатываются системой событий Windows Forms и преобразуются в соответствующие игровые реакции.

Хранение реализуется в локальной файловой системе пользователя в директории приложения. Формат хранения — текстовый, позволяющий легко редактировать и считывать данные средствами C++.

Игровое приложение должно выполнять следующие функции:

* загрузка интерфейса и начальных данных при запуске;
* отображение меню с возможностью запуска игры и справки;
* запуск игрового цикла с отображением объявлений, подсчётом очков и определением уровня ваших знаний;
* динамическое изменение сложности в зависимости от счёта;
* управление звуком.
  1. **Инструменты разработки**

Разработка программного продукта осуществляется в интегрированной среде Microsoft Visual Studio Community Edition — одной из наиболее популярных и мощных IDE для создания Windows-приложений. Данный выбор обусловлен рядом факторов, в том числе:

* ­наличие встроенного конструктора форм, позволяющего визуально проектировать интерфейс без необходимости ручного позиционирования элементов;
* интеграция с системой отладки, профилирования и анализа кода, значительно упрощающая процесс устранения ошибок;
* поддержка C++/CLI, что предоставляет доступ к мощным возможностям .NET и библиотеке Windows Forms;
* удобная система автоматического дополнения кода (IntelliSense), повышающая скорость и точность программирования;
* широкая распространённость и наличие документации, примеров, сообществ поддержки и обучающих материалов.
* высокие мощности Visual Studio community

Visual Studio обеспечивает гибкость настройки проекта, возможность подключения сторонних библиотек, быструю компиляцию, отладку и управление версионностью проекта.

Выбранный язык реализации — C++/CLI, предоставляющий синтаксическую совместимость с C++ и в то же время интеграцию с .NET. Это позволяет разрабатывать приложения с графическим интерфейсом, использовать компоненты .NET, а также обеспечить более простую работу с памятью за счёт использования сборщика мусора.

Технологический стек:

* Windows Forms — визуальный интерфейс;
* стандартные библиотеки C++ и .NET Framework;

Также для разработки программы необходимы:

* Word 2016 – необходим для написания пояснительной записки;
* Adobe Photoshop CC 2021 – необходим для обработки фотографий.
  1. **Требования к приложению**

В рамках реализации приложения предусмотрены следующие ограничения:

* программа функционирует исключительно в среде Windows;
* отсутствие поддержки мобильных и веб-платформ;
* игровое приложение рассчитано на одного пользователя (однопользовательский режим);
* нет функции сетевого взаимодействия и обмена результатами;
* все игровые процессы происходят в рамках одного окна фиксированного размера;
* результаты и данные сохраняются исключительно локально, без использования облачных хранилищ.

При разработке интерфейса приложения преимущественного использованы бирюзовые и желтые оттенки. Основные разделы приложения доступны с первой страницы после загрузки. Каждое окно проекта имеет ясную визуальную иерархию своих элементов. Разработан понятный пользователю интерфейс, фрагменты текста располагаются на экране так, чтобы пользователь не испытывал какого-либо дискомфорта в плане восприятия информации, отображённой на экране.

Минимальные системные требования:

* процессор: Intel Pentium Dual-Core или аналогичный AMD;
* ОЗУ: от 2 ГБ;
* диск: не менее 300 МБ свободного места;
* ОС: Windows 7 SP1 и выше.

Рекомендуемые системные требования:

* процессор: Intel Core i5 / AMD Ryzen 3 и выше;
* ОЗУ: от 4 ГБ;
* диск: SSD или HDD с не менее 300МБ свободного места;
* ОС: Windows 10 / Windows 11 с актуальными обновлениями.

Для повышения стабильности и защищённости работы приложения реализованы следующие меры:

* + проверка наличия и структуры файлов при загрузке/сохранении данных;
  + обработка исключений при работе с файлами и взаимодействии с пользователем;
  + предотвращение повторной активации кнопок (например, двойного запуска игры);
  + валидация значений, вводимых или генерируемых в ходе игры (например, защита от отрицательных очков);

**2 Проектирование задачи**

**2.1 Организация данных**

Разработка программного продукта, представляющего собой игровое приложение «Автобаза», требует продуманной организации данных как на логическом, так и на физическом уровне. Для эффективной реализации игры в среде Windows Forms с использованием языка C++/CLI необходимо разделить все информационные потоки на три ключевые категории: входная, выходная и условно-постоянная информация.

Входная информация

Входная информация представляет собой события, инициируемые пользователем в ходе взаимодействия с программой. Основным способом ввода данных является использование мыши, а именно:

* щелчки по кнопкам (игровой процесс);
* взаимодействие с кнопками меню;
* переключение настроек (например, звук вкл./выкл.);
* сброс или выход из приложения.

На программном уровне все входные события обрабатываются через стандартные обработчики событий Windows Forms, такие как Click, MouseDown, MouseEnter и т. д. В логической структуре такие события могут быть представлены как отдельные методы-контроллеры, привязанные к конкретным объектам UI. Поскольку данные не накапливаются в виде массивов, они обрабатываются мгновенно, без промежуточного хранения.

Выходная информация

Выходные данные являются результатом обработки пользовательского взаимодействия. Ключевыми выходными компонентами являются:

* текущий баланс;
* визуальные элементы, отображающие автомобили;
* сообщения о завершении игры;
* аудио сопровождение.

Для реализации вывода используется как непосредственное обновление свойств визуальных компонентов (например, Label.Text), так и вызов методов для управления мультимедиа. Числовые данные (баланс) хранятся в переменных типа int, а затем преобразуются в строку для вывода на экран.

**2.2 Процессы**

Основу логики функционирования игрового приложения составляет игровой цикл, в котором реализуются все ключевые этапы взаимодействия между пользователем и программой. Описание процессов даёт представление о механизмах, лежащих в основе игрового поведения, а также о структурной взаимосвязи всех компонентов проекта.

Алгоритм игрового процесса начинается с запуска приложения, что сопровождается отображением стартовой формы загрузки, также известной как loading screen. Эта форма представляет собой кратковременное вступительное окно, которое служит своеобразной заставкой — она может содержать логотип игры, её название, а также краткую анимацию или звуковое сопровождение. Основная цель этой стадии — создать у пользователя ощущение готовности к началу работы, а также скрыть от него этапы внутренней подготовки интерфейса и логики.

Как только завершены все внутренние процессы подготовки, форма загрузки скрывается, и пользователь автоматически перенаправляется на главную форму меню. Главное меню является центральной точкой навигации, через которую осуществляется доступ ко всем ключевым функциям приложения. Здесь пользователю предлагаются базовые опции: начать игру, обратиться к справке, узнать об авторе, правилах игры или завершить работу приложения. Все элементы интерфейса выполнены в едином визуальном стиле.

После выбора опции "Играть" начинается фактический игровой процесс. Программа переходит в активный игровой режим: все параметры игры сбрасываются до стартовых значений, включая возврат стартового баланса, сброс таймера и генерацию стартовой позиции автомобилей.

Во время активного игрового цикла на экране появляются автомобили, визуализированные в виде изображений. Каждый автомобиль появляется в центральной части экрана. Пользователь, используя мышь, должен быстро среагировать и выбрать автомобиль и кликнуть на кнопки. При правильном выборе игрок получает внутриигровую валюту, при неправильном выборе автомобиля – теряет внутриигровую валюту.

Когда игра завершается, игра автоматически завершает игровую сессию, отображая экран с результатом. Здесь показываются баланс внутриигровой валюты. После завершения игры пользователь может начать новую попытку или выйти в главное меню. Этот цикл можно повторять неограниченное количество раз, что делает игру пригодной для краткосрочных сессий, тренировок на развитие мышления и просто развлечения.

**2.3 Описание внешнего пользовательского интерфейса**

После запуска игрового приложения пользователь сначала видит форму загрузки, на которой происходит инициализация всех компонентов. Затем автоматически открывается главное меню. В главной форме присутствует панель меню(MenuStrip) сверху, содержащая пункт «меню», при выборе которого отображается контекстное меню. В этом меню доступны следующие опции: «Играть» (запускает игровую форму), «Об авторе» и «Правила» (также реализованы в виде отдельных кнопок на форме, при нажатии на которые отображаются соответствующие панели с информацией), «Звук» (включение или отключение звукового сопровождения), а также «Выйти» для завершения работы программы. Все элементы интерфейса организованы логично и лаконично: переключение между игровым режимом и информационными панелями происходит без смены окна.

После перехода на игровую форму игроку доступна кнопка «В меню», прерывающая игру, возвращая игрока назад в главное меню

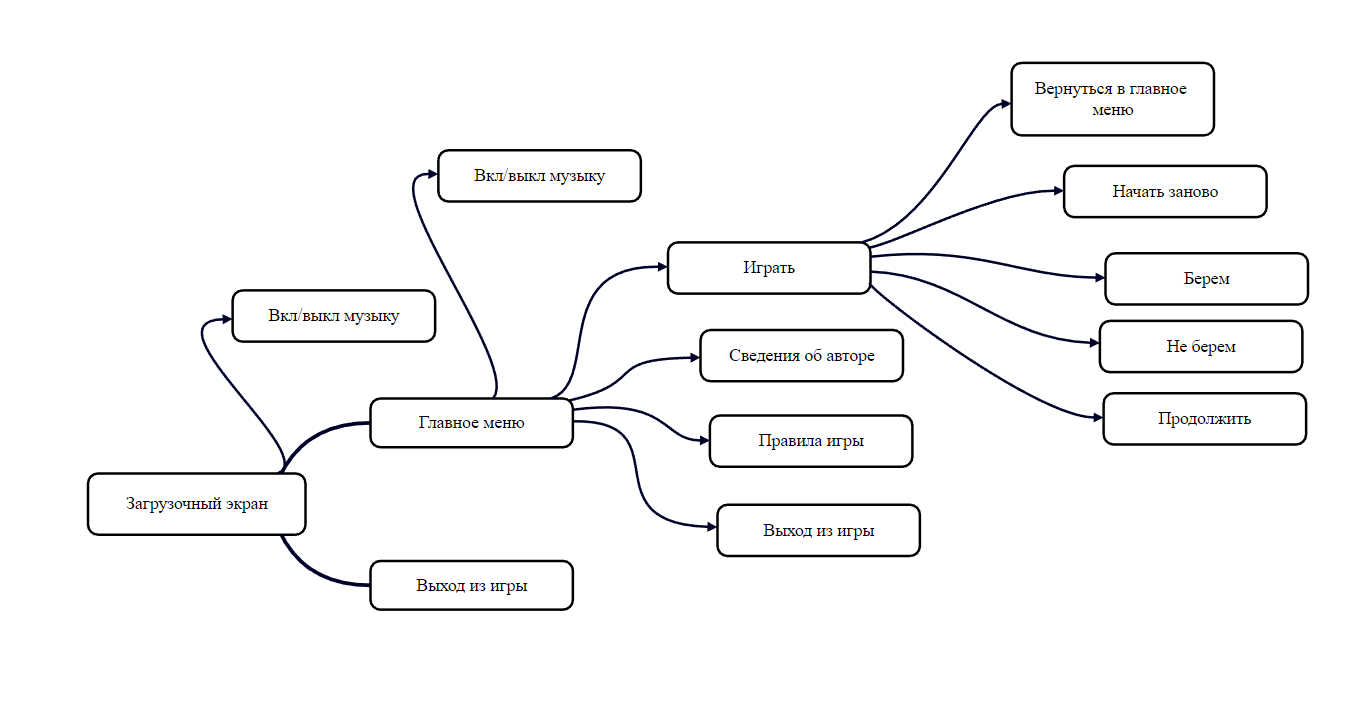
Структура навигации по проекту представлена на рисунке 1:

Рисунок 1 – Навигация по проекту

**3 Реализация**

**3.1 Реализация проекта**

**3.1.1 Структура программы**

Данный курсовой проект содержит 3 модуля. Далее рассмотрим назначение каждого модуля:

Модуль MyForm1 – содержит отображения формы приветствия. С него идет переход на главное меню;

В модуле MyForm2 находится главное меню, информация об авторе, правила, справка и отключение звука с него осуществляется переход на другие модули. С него идет переход в игру;

Модуль MyForm содержит в себе саму игру, кнопки(утки), графические элементы игры, кнопку отключения звука и возвращения в главное меню.

**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 |
| 1 | 2 | 3 | 4 |
| private: System::Void MyForm1\_Load(System::Object^ sender, System::EventArgs^ e) { | MyForm1 | MyForm1 | Запуск таймера1 |
| private: System::Void timer1\_Tick(System::Object^ sender, System::EventArgs^ e) { | MyForm1 | Timer1 | Реализация анимации загрузки |
| private: System::Void button4\_Click(System::Object^ sender, System::EventArgs^ e) { | MyForm2 | Button4 | Выход из приложения |
| private: System::Void button3\_Click(System::Object^ sender, System::EventArgs^ e) { | MyForm2 | Button3 | Открытие панели «правила» |
| private: System::Void button1\_Click(System::Object^ sender, System::EventArgs^ e) { | MyForm2 | Button1 | Переход на форму с игрой |

Продолжение таблицы 1

|  |  |  |  |
| --- | --- | --- | --- |
| 1 | 2 | 3 | 4 |
| private: System::Void pictureBox2\_Click(System::Object^ sender, System::EventArgs^ e) { | MyForm2 | pictureBox2 | Включение/выключение музыки |
| private: System::Void button2\_Click(System::Object^ sender, System::EventArgs^ e) { | MyForm | Button2-37 | Зачисление или счисление денег с баланса, переход к следующему автомобилю |
| private: System::Void MyForm\_Load(System::Object^ sender, System::EventArgs^ e) { | MyForm | MyForm | Содержит все фотографии |
| private: System::Void pictureBox14\_Click(System::Object^ sender, System::EventArgs^ e) { | MyForm | pictureBox14 | Включение/выключение музыки |
| СправкаToolStripMenuItem\_Click() | MyForm2 | MenuStrip | Справка |

**3.1.3 Описание использованных компонентов**

Описание использованных при разработке компонентов необходимо привести в виде следующей таблицы:

Таблица 2 – Использованные компоненты

|  |  |  |
| --- | --- | --- |
| Компонент | На какой форме расположен | Назначение |
| Label | Все формы | Отображение заголовков и текста |
| PictureBox | Все формы | Отображение графической части |
| Panel | MyForm2 | Группировка интерфейсных элементов, отображение информации |
| WindowsMediaPlayer | MyForm1, MyForm2 | Воспроизведение звуковых элементов |
| Button | MyForm1, MyForm2 | Запуск действий: скрытие/отображение панелей с информацией, переходы, управление |
| MenuStrip | MyForm1 | Главное меню («Справка») |

**3.2 Спецификация программы**

Точное название проекта и его состав приводится в таблице 3.

Таблица 3 – Спецификация программы

|  |  |
| --- | --- |
| Имя файла | Назначение |
| 1 | 2 |
| kursachok.exe | Исполняемый файл проекта |
| kursachok.sln | Решение проекта Visual Studio |
| MyForm1.cpp / MyForm1.h | Код и заголовок загрузочной формы |
| MyForm2.cpp / MyForm2.h | Модуль главного меню |
| MyForm.cpp / MyForm.h | Модуль игры |
| Resources.resx | Файл ресурсов проекта |
| меню.png | Изображения меню |
| 1.png, 2.png, 3.png, 4.png, 5.png, 6.png, 7.png, 8.png, 9.png, 10.png, 11.png, 12.png, 13.png. | Изображения автомобилей |
| загрузка.png, загрузка1.png | Изображения загрузочного экрана |
| bestrec.txt, lastrec.txt | Файлы сохранений рекордов |
| game.mp3, menu.mp3, shoot.mp3 | Файлы игровых звуков и фоновой музыки |
| бабки.png | Изображение баланса |
| музыкан.wav | Звук |
| soundoff.png, soundon.png | Изображения кнопки включения |
| help.chm | Файл справки |

**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.

Отчет о результатах тестирования приведён в данной таблице:

Таблица 4 – Отчёт о результатах тестирования

|  |  |  |  |
| --- | --- | --- | --- |
| Название теста | Ожидаемый результат | Результат  тестирования | Фактический результат |
| Нажатие клавиши после загрузки | Переход на форму главного меню | Переход на форму главного меню | Выполнено |
| Проверка изображений авто | Рабочая изображение | Рабочая изображение | Выполнено |
| Проверка кнопки «играть» | Переход на форму с игрой | Переход на форму с игрой | Выполнено |
| Проверка кнопки «Звук» | Включение/ Выключение звука | Включение/ Выключение звука | Выполнено |
| Проверка пункта MenuStrip«Об авторе» | Отображение «об авторе» | Отображение «об авторе» | Выполнено |
| Проверка кнопки «Правила» | Отображение «Правила» | Отображение «Правила» | Выполнено |
| Проверка кнопки «Выйти» | Выход из приложения | Выход из приложения | Выполнено |
| Проверка пункта MenuStrip «Справка» | Открытие справки | Открытие справки | Выполнено |
| Проверка зачисления на баланс | Зачисление валюты на баланс | Зачисление валюты на баланс | Выполнено |
| Проверка счисление с баланса | Счисление валюты с баланса | Счисление валюты с баланса | Выполнено |
| Проверка кнопки «в меню» | Возвращение на форму главного меню | Возвращение на форму главного меню | Выполнено |

Все этапы тестирования подтвердили, что Игровое приложение «Автобаза» работает стабильно. Элементы управления — новый автомобиль, выбор действий — функционируют согласно сценарию. Логика игры отлажена: она точно реагирует на действия игрока, корректно завершается и переходит к нужным этапам. В результате игрок не столкнётся с неожиданными сбоями или недоработками.

**5 Руководство пользователя**

* 1. **Общие сведения о программном продукте**

Название приложения: Автобаза

Назначение: Игра на проверку знаний в области автомобилей, в которой игроку необходимо выбрать авто приносящее большую выгоду при перепродаже, соревнуясь с самим собой.

Основные возможности:

* запуск игрового режима с автомобилями;
* подсчёт и отображение баланса;
* просмотр справочной информации и сведений об авторе;
* управление звуком;

Характеристики:

* интерфейс: Windows Forms (.NET), язык интерфейса — русский;
* оффлайн-режим, полностью локальное приложение;
* удобное управление: мышь;
* адаптировано под широкоформатные дисплеи;
* поддержка визуальной обратной связи, диалоговых окон и панели навигации.

Областью применения являются анализ рынка и принятие решений (навыки торговли и управления финансами)

Периодичность использования не ограничена

Среда функционирования – ОС windows(7,8,10,11)

* минимальные требования:
* процессор: 1 ГГц
* оперативная память: 2 ГБ
* свободное место на диске: 200 МБ
* разрешение экрана: не менее 1400×800 пикселей

Рекомендуемые параметры:

* процессор: 2 ГГц+
* ОЗУ: 2 ГБ и более
* SSD накопитель
* разрешение экрана: 1920×1080 или выше
  1. **Установка программы**

Для установки игрового приложения «Автобаза» пользователю необходимо запустить исполняемый файл kursachok.exe. После запуска откроется диалоговое окно установки, в котором следует выбрать путь установки и следовать подсказкам установщика.

После завершения установки программа готова к использованию. Для запуска можно использовать ярлык, созданный на рабочем столе, либо открыть исполняемый файл вручную через установленную директорию.

* 1. **Выполнение программы**

**5.3.1 Запуск программы**

Игрового приложение «Автобаза» можно запустить из каталога, в который было установлено игровое приложение. Либо запустив kursachok.exe, либо через ярлык, созданный на рабочем столе.

**5.3.2 инструкция по работе с программой**

После запуска происходит следующая последовательность действий:

* экран загрузки (рисунок 2). При запуске приложения отображается загрузочный экран, где происходит инициализация компонентов.

Рисунок 2 – Загрузка

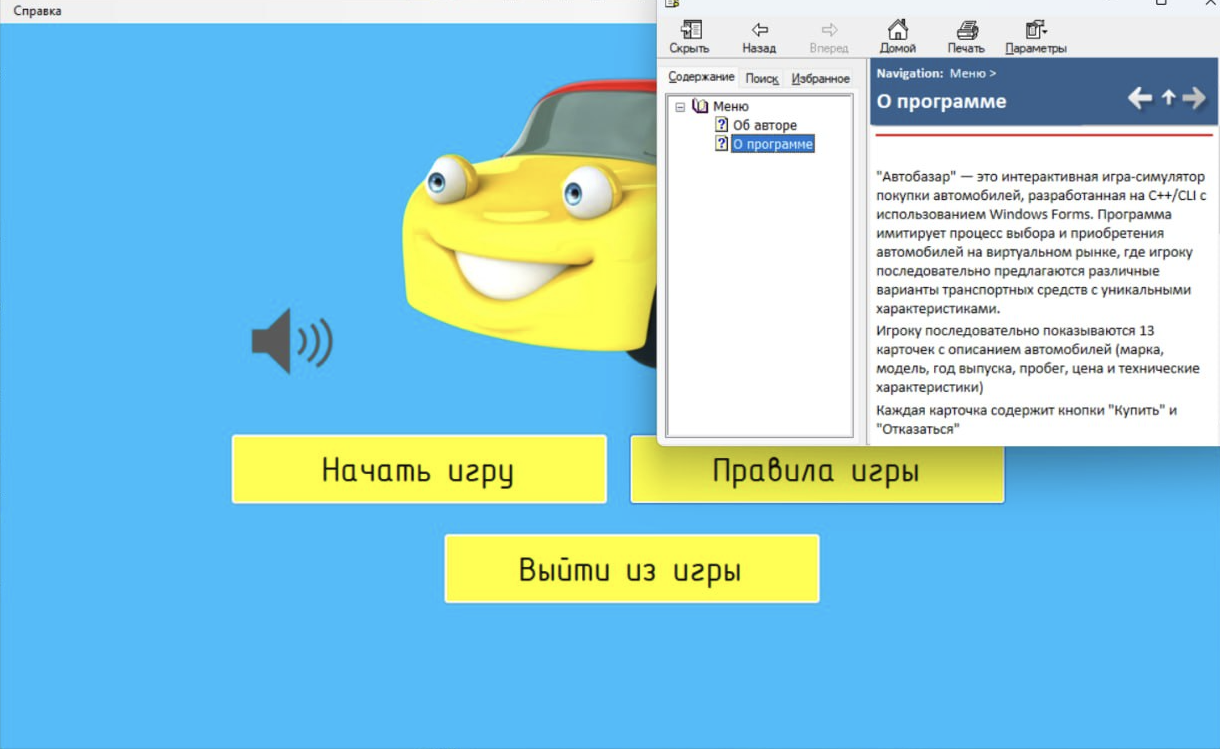
* главное меню (рисунок 3) содержит в себе спокойную зацикленную музыку и элементы управление, такие как: кнопки, MenuStrip, панели и т.д.. Тут игрок может отключить/включить звука, перейти в саму игру, прочитать правила игры, либо выйти из приложения.

Рисунок 3 – Главное меню

* на главной форме реализованы кнопки и пункты в MenuStrip под названиями: «О прогрмамме» и «Об авторе».

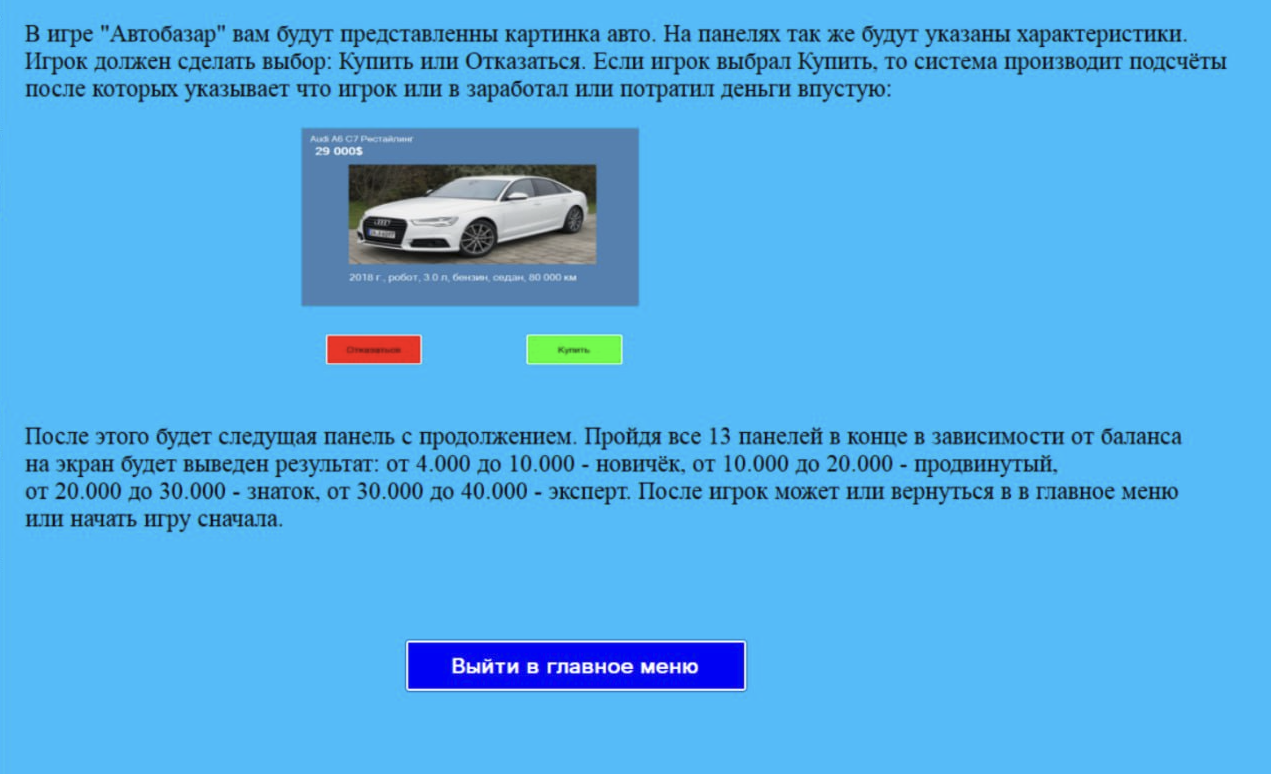
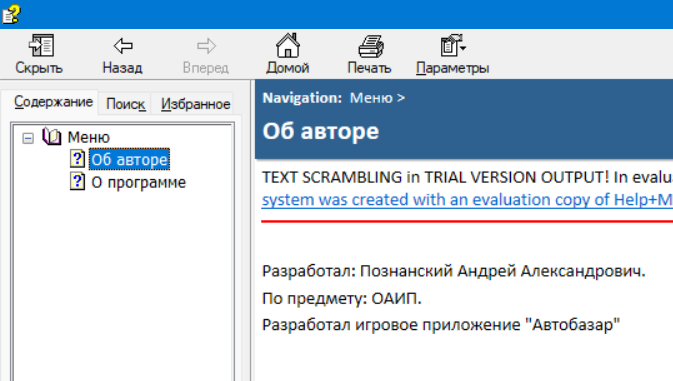
Рисунок 4 – Об авторе

Рисунок 5 - Правила

* в игре необходимо выбирать наиболее выгодную машину. По завершению выбора будет виден уровень ваших знаний в области продажи автомобилей. Затем – игра заканчивается и можно осуществить переход в главное меню или начать заново. Пример игровой формы можно увидеть на рисунке 6

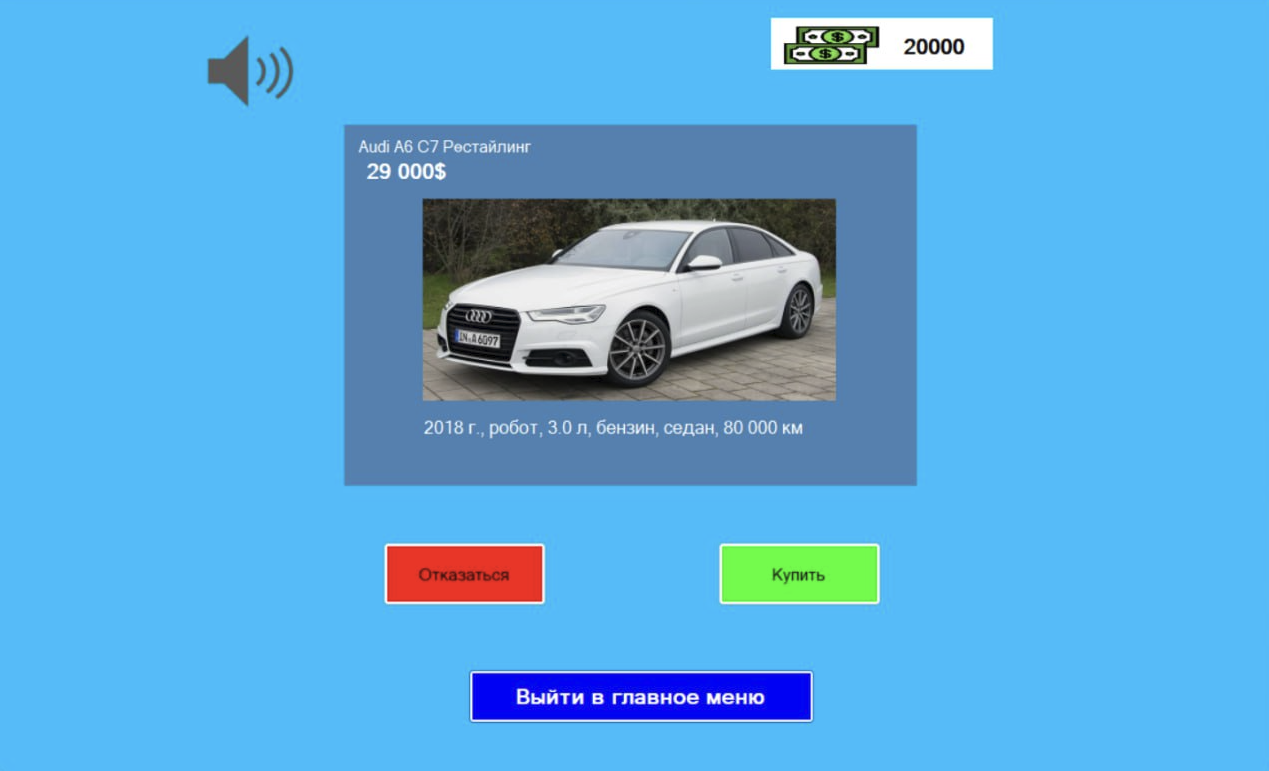


Рисунок 6 – Пример игры

**5.3.3 Завершение работы с программой**

Закрытие приложения осуществляется:

* кнопкой «Выйти» в главном меню (рисунок 3)
* через “Alt + F4”

**5.4 Использование системы справочной информации**

Справочную систему можно запустить с помощью пункта меню «Справка» (рисунок 7)

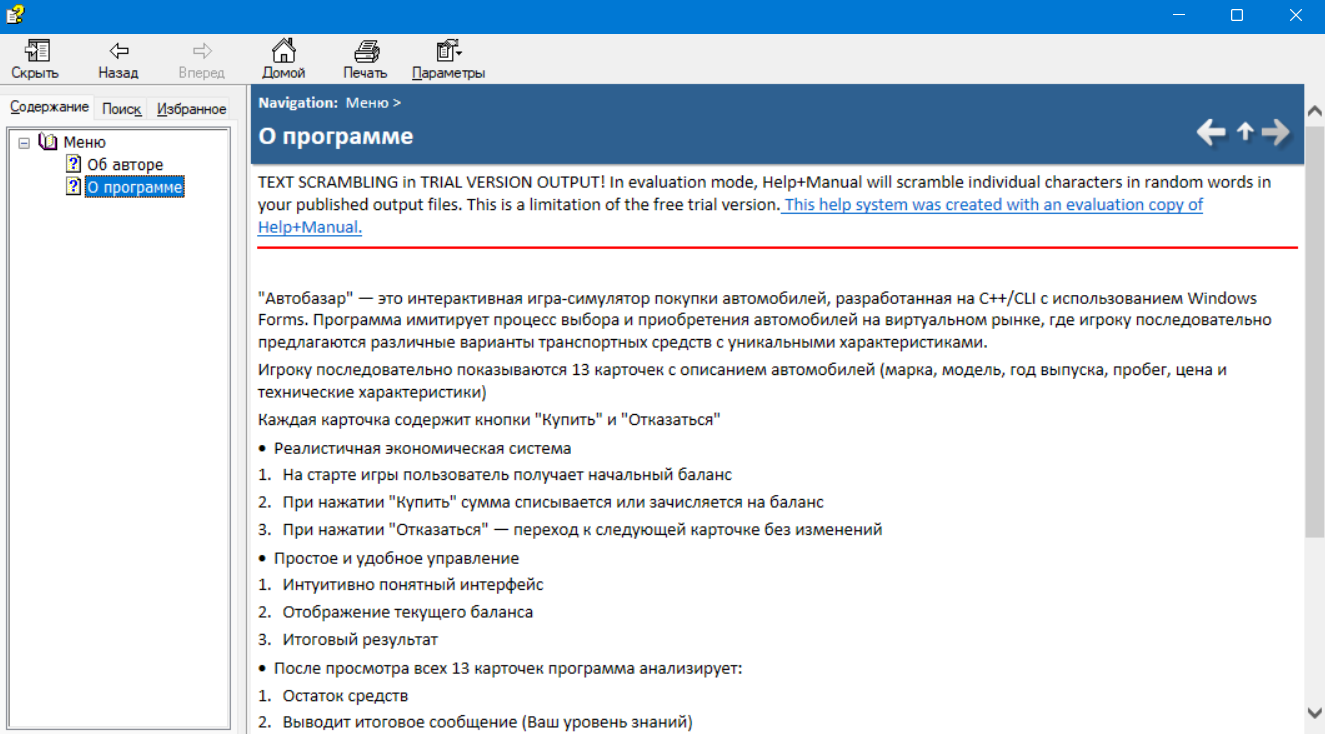


Рисунок 7 – Справка

**Заключение**

В рамках проектной работы была поставлена задача разработки настольного игрового приложения «Автобазар», ориентированного на тренировку сообразительности и финансовой грамотности пользователя. Основной целью являлось создание интуитивно понятного и функционального интерфейса, а также реализация игрового процесса, визуальную обратную связь и возможность работы без подключения к интернету.

Поставленная задача была успешно реализована. Разработано полнофункциональное приложение с главным меню, формой игры, формой загрузки и вспомогательными интерфейсными элементами, такими как панели «Об авторе» и «Правила», а также встроенные средства взаимодействия с пользователем. В приложении корректно реализована навигация между окнами, а также управление звуковым сопровождением.

В процессе реализации использовались средства среды разработки Visual Studio на платформе .NET Framework с использованием Windows Forms, что позволило быстро и удобно разрабатывать интерфейс и обрабатывать события. Также были применены объектно-ориентированные подходы к проектированию, включая разделение логики на классы и работа с событиями. Визуальные элементы и механика взаимодействия с пользователем опирались на стандартные компоненты WinForms с адаптацией под минимальное разрешение экрана 1400×800 пикселей.

Проектные решения в полной мере соответствуют изначально сформулированному техническому заданию. Все заявленные функции — от запуска игры и отображения рекордов до справочной информации и интерфейса управления — были реализованы и протестированы. Несоответствий между реализацией и изначальными требованиями не выявлено.

Проект обладает высокой гибкостью и возможностью дальнейшей модификации. При необходимости могут быть добавлены новые игровые режимы, уровни сложности, а также введена система пользовательских аккаунтов и таблица лидеров. Кодовая структура поддерживает масштабирование, благодаря чему приложение может быть адаптировано как под другие типы игр, так и под обучение, тренажёры или тестовые приложения.

**Список использованных источников**

1. MSDN Microsoft Docs – Документация по Windows Forms [Электронный ресурс]. - Режим доступа: <https://learn.microsoft.com/ru-ru/dotnet/desktop/winforms/>. – Дата доступа: 21.06.2025
2. LearnCpp.com – Современный самоучитель по C++ [Электронный ресурс]. - Режим доступа: <https://www.learncpp.com/>. – Дата доступа: 22.06.2025
3. Cppreference – Онлайн-справочник по языку C++ [Электронный ресурс]. - Режим доступа: <https://en.cppreference.com/>. – Дата доступа: 22.06.2025