

1 Анализ задачи

1.1 Постановка задачи

Наименование задачи: Разработка игрового приложения «Змейка».

Задача данного проекта заключается в создании игрового приложения «Змейка».

С момента своего появления игра «Змейка» завоевала популярность благодаря простому, но увлекательному игровому процессу. Игра стала культовой, и многие поколения игроков наслаждались ею на различных платформах.

Игровое приложение «Змейка» будет представлять собой классическую игру, в которой игрок управляет змейкой, собирающей бонусы и избегающей препятствий. Цель игры — набрать как можно больше очков, не сталкиваясь с собственным хвостом или препятствиями.

Игра будет включать различные уровни сложности, а также настройку параметров, таких как скорость змейки. Игровой процесс будет сопровождаться звуковыми эффектами, а сложность игры будет нарастать по мере продвижения.

Цель разработки: Создание развлекательного игрового приложения, которое будет способствовать развитию логического мышления, мотивации к изучению разработки игр и предоставлению пользователям увлекательного досуга.

Назначение: Приложение предназначено для развлечения, развития логического мышления и демонстрации реализации игрового приложения с графическим представлением посредством использования возможностей графических библиотек.

Предметная область: Классическая игра «Змейка». На основе этой концепции будет построена и реализована модель игры.

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

Источники и способы получения данных: Проект будет содержать несколько уровней сложности с генерируемыми препятствиями и бонусами. Игровой процесс будет сопровождаться звуковыми эффектами и нарастающей сложностью.

Функциональные возможности разрабатываемого программного продукта:

- Начало новой игровой сессии.
- Настройка параметров игры (скорость змейки, уровень сложности).
- Просмотр таблицы рекордов.

					КП 2-40 01 01.33.41.06.24 ПЗ	Лист
Изм.	Лист	№ докум.	Подпись	Дата		6

1.2 Инструменты разработки

Для разработки данного проекта выбрана среда Delphi, поскольку это удобная среда для разработки игр, поддерживающая множество платформ и предоставляющая обширные возможности для визуализации и логики игры.

Также для разработки программы необходимы:

– Adobe Photoshop (для создания и редактирования игровых изображений и текстур) – это популярная и широко используемая программа для работы с растровыми изображениями, разработанная компанией Adobe Inc. Она предоставляет обширный набор инструментов для создания, редактирования и ретуширования фотографий и цифровых изображений. Adobe Photoshop применяется как профессиональными дизайнерами и фотографами, так и любителями для разнообразных задач.

– Google Chrome (для поиска информации и ресурсов (графика, звуки)) – это бесплатный веб-браузер, разработанный компанией Google. Он был выпущен в 2008 году и с тех пор стал одним из самых популярных браузеров в мире благодаря своей скорости, простоте и множеству полезных функций.

– Microsoft Word (для написания пояснительной записки) – это популярная программа для обработки текстов, разработанная компанией Microsoft. Она входит в состав офисного пакета Microsoft Office и используется для создания, редактирования и форматирования текстовых документов.

– Microsoft PowerPoint (для создания отчетной презентации) – это популярная программа для создания презентаций, также разработанная компанией Microsoft. Она входит в состав офисного пакета Microsoft Office и используется для создания визуальных слайд-шоу, которые могут включать текст, изображения, графику, видео и аудио.

Характеристики компьютера, использованного для разработки:

- Процессор: Intel Core i5 11400H.
- ОЗУ: 16 GB.
- Память: SSD 512 GB.
- ОС: Windows 11.

1.3 Требования к приложению

На этапе исследования предметной области был установлен целый ряд требований, предъявляемых к разрабатываемой задаче.

Требования к интерфейсу:

- Интуитивно понятный интерфейс и управление;

					КП 2-40 01 01.33.41.06.24 ПЗ	Лист
Изм.	Лист	№ докум.	Подпись	Дата		7

- Небольшое количество информации на игровом экране;
- Ясная визуальная иерархия элементов интерфейса;
- Визуально приятный интерфейс с яркими и четкими элементами;
- Наличие информации о правилах игры;
- Удобные для управления игровые элементы.

Минимальные системные требования:

- Процессор: 1.8 GHz;
- ОЗУ: 1 GB;
- От 50 MB доступного места на жестком диске.

Рекомендуемые системные требования:

- Двухъядерный процессор Intel или аналогичный AMD;
- ОЗУ: 2 GB;
- От 50 MB доступного места на жестком диске.

При разработке интерфейса приложения:

- Использованы преимущественно яркие и контрастные оттенки;
- Основные разделы приложения доступны с главного меню;
- Каждое окно проекта имеет ясную визуальную иерархию своих элементов;
- Понятный пользователю интерфейс, элементы управления удобно расположены и легко доступны;
- Размеры игровых объектов (змейки и бонусов) удобны для управления и игры.

Ошибки программы вследствие некорректного взаимодействия пользователя при работе с программой через графический интерфейс не должны влиять на общую работоспособность приложения.

Надежность программы должна быть на высоком уровне, не допуская ситуаций программных ошибок и некорректной работы приложения.

					КП 2-40 01 01.33.41.06.24 ПЗ	Лист
Изм.	Лист	№ докум.	Подпись	Дата		8

2 Проектирование задачи

2.1 Организация данных

Проектирование задачи – это очень важный и ответственный этап в разработке любого приложения, включая игровое приложение "Змейка". Методы, посредством которых пользователь управляет игрой, строятся 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 три вида данных.

Первым видом данных являются те, которые будут введены разработчиком на этапе реализации задачи. Сюда можно отнести изображения (иконки), звуковые файлы, описание логики игры, исходные коды для алгоритмов управления.

Вторым видом данных, используемых в программе, является вводимая пользователем информация. Входной информацией в разрабатываемой программе будут являться данные, которые изменяются пользователем во время игры: направление движения змейки (вверх, вниз, влево, вправо). Эта информация будет вводиться с клавиатуры или сенсорных жестов.

Третьим видом данных является результат программы – текущее состояние игры. Это включает в себя положение змейки, позиции еды на игровом поле, текущий счет пользователя и состояние игры (например, активна ли игра или завершена). Эти данные постоянно обновляются во время выполнения игры.

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

2.2 Процессы

Согласно всем перечисленным требованиям и указаниям, которые были рассмотрены в разделе «Анализ задачи», было определено, чем конкретно должна заниматься разрабатываемая курсовая программа. Главной её задачей будет являться: создание и управление процессом игры "Змейка". Весь данный процесс будет заключаться в следующем.

Пользователь управляет змейкой, которая перемещается по игровому полю. В зависимости от действий пользователя, змейка будет двигаться в выбранном направлении. Когда змейка съедает еду, её длина увеличивается, и на игровом поле появляется новый кусок еды. Если змейка сталкивается с границей игрового поля или с самой собой, игра завершается.

					КП 2-40 01 01.33.41.06.24 ПЗ	Лист
						9
Изм.	Лист	№ докум.	Подпись	Дата		

Для реализации поставленной задачи будут использованы процедуры. Например, с помощью процедур будет осуществляться управление движением змейки, генерация еды и обработка столкновений.

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, что от него требуется.

Для организации эффективной работы пользователя нужно создать целостное приложение, в котором все компоненты 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):

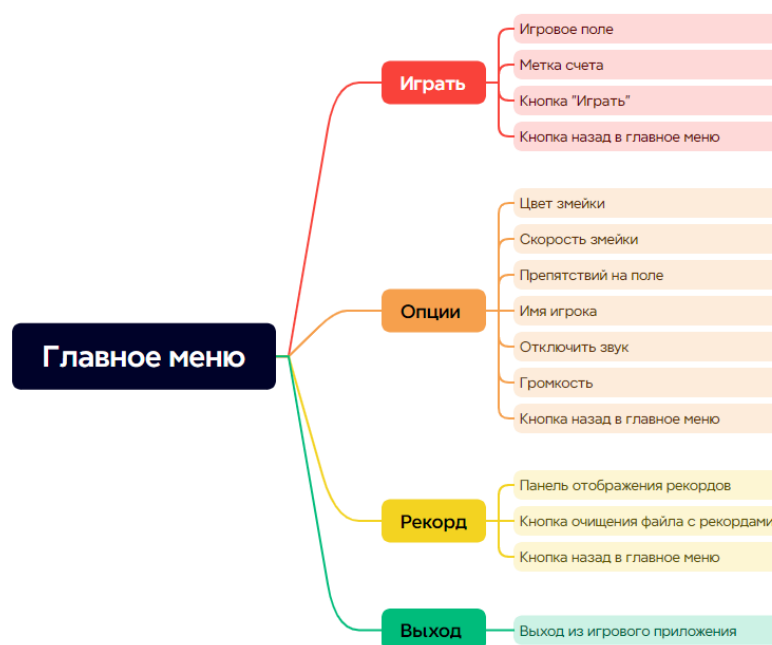


Рисунок 1 – Навигация между окнами программы

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 Логотипа (Рисунок 2);
- 2 Заставки (Рисунок 3);
- 3 Главного окна (Рисунок 4);
- 4 Окна, содержащее игру (Рисунок 5);
- 5 Окна, содержащее настройки игры (Рисунок 6);
- 6 Окна, содержащее рекорды игрока (Рисунок 7).

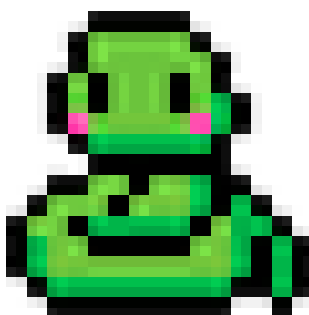


Рисунок 2 – Прототип логотипа

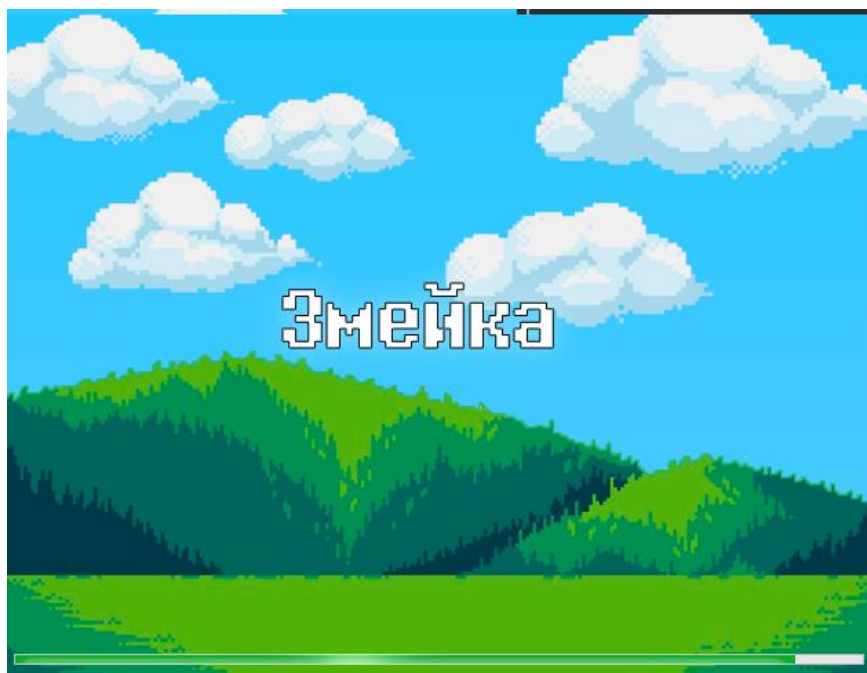


Рисунок 3 – Прототип заставки

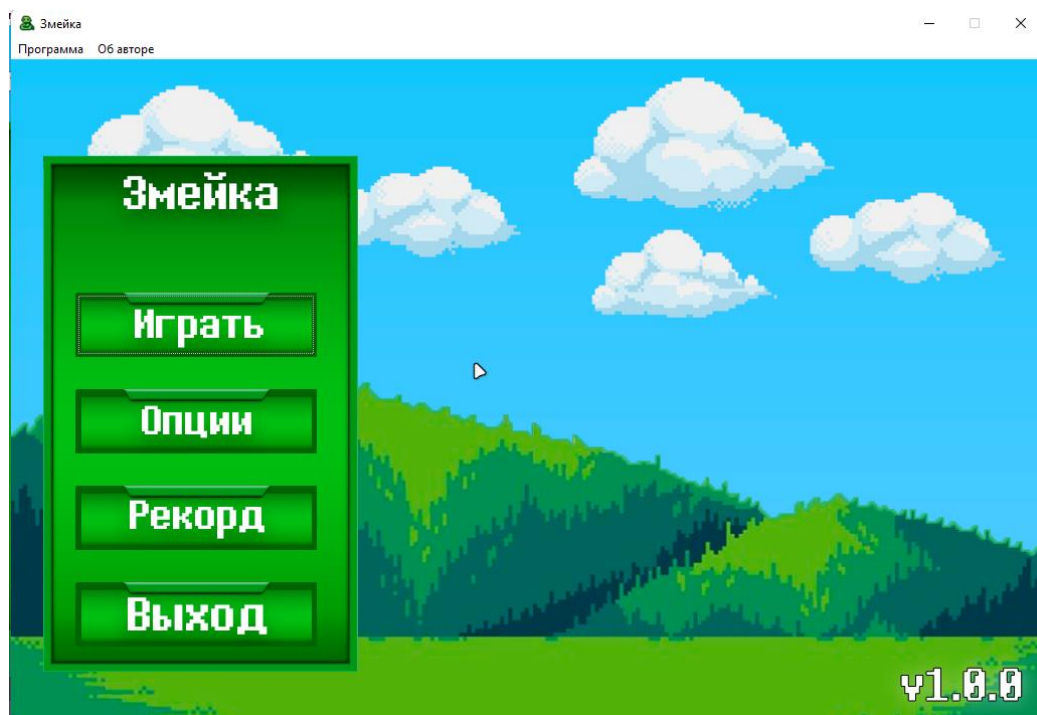


Рисунок 4 – Прототип главного окна



Рисунок 5 – Прототип окна, содержащее игру

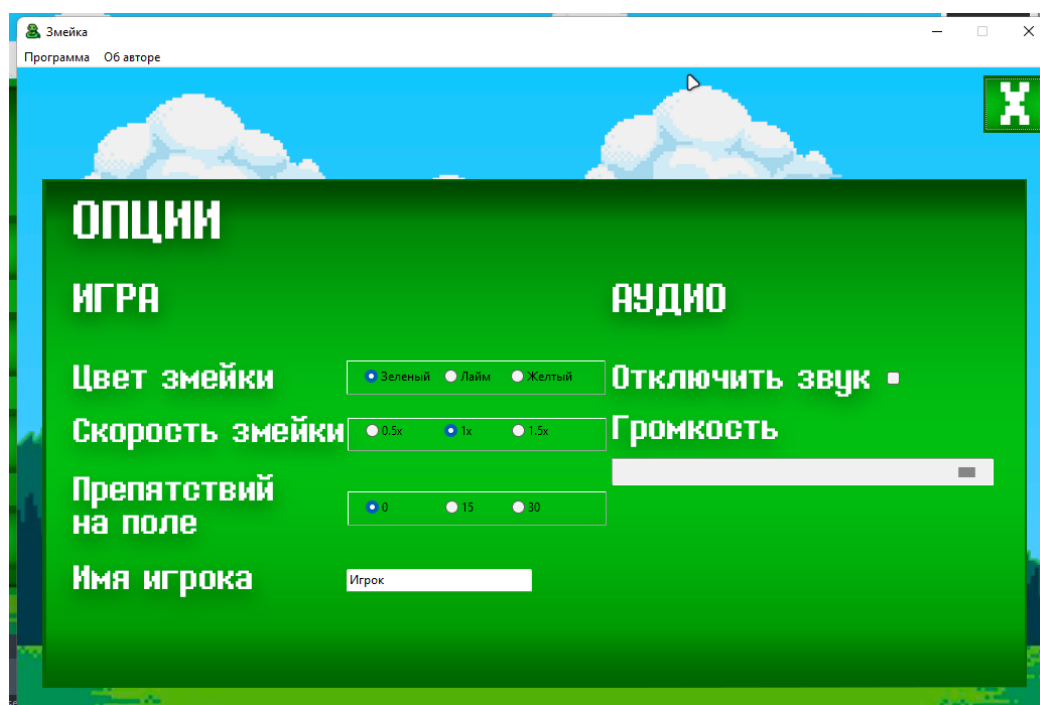


Рисунок 6 – Прототип окна, содержащее настройки игры

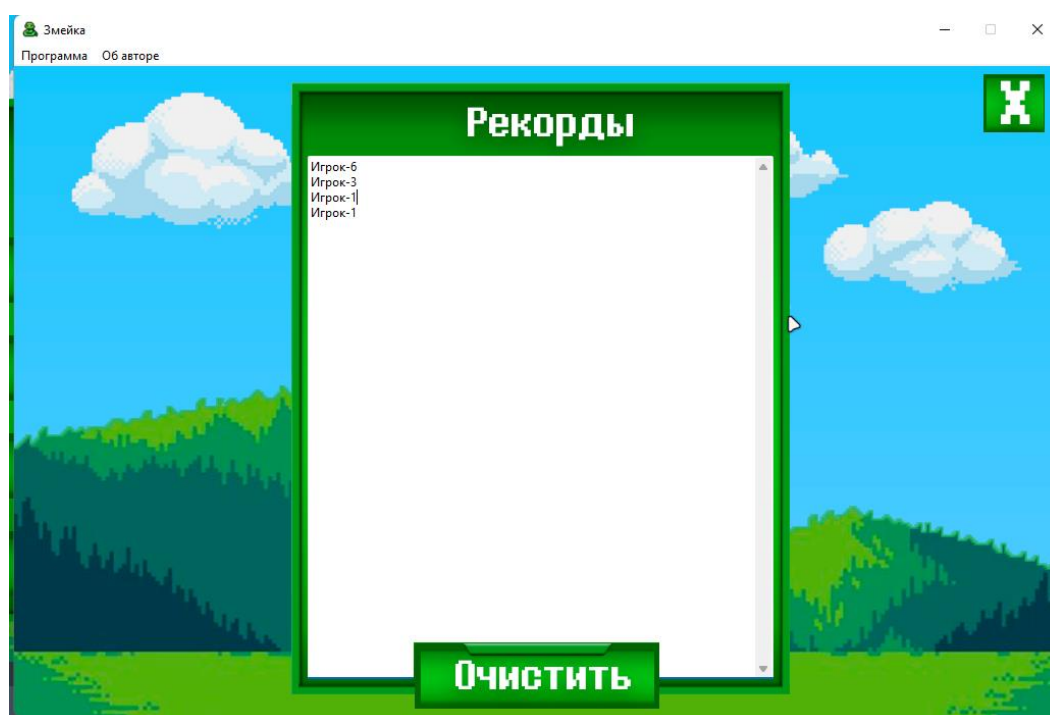


Рисунок 7 – Прототип окна, содержащее рекорды игрока

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

3 Реализация

3.1 Структура программы

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

1 Модуль Unit1 – отвечает за главное окно приложения и обеспечивает навигацию между различными частями программы;

2 Модуль Unit2 – отвечает за отображение экрана загрузки, который отображается при запуске приложения;

3 Модуль Unit3 – отвечает за реализацию логики игры "Змейка". Он включает в себя отрисовку игрового поля, управление змейкой, генерацию пищи и препятствий, обработку столкновений и управление игровыми состояниями;

4 Модуль Unit4 – отвечает за настройки и управление звуком в игре, предоставляя интерфейс для пользовательских настроек и управления звуковым сопровождением;

5 Модуль Unit5 – отвечает за отображение и управление рекордами игры. Он предоставляет интерфейс для просмотра, очистки и сортировки рекордов, сохранённых в файле «snake_record.txt»;

3.1.1 Структура и описание процедур и функций пользователя

Описание разработанных процедур находятся в таблице 1.

Таблица 1 – Процедуры

Полное имя процедуры (функции)	В каком модуле находится	За каким компонентом закреплена	Назначение
TForm1.BitBtn1Click	Unit1	BitBtn1	Скрывает форму Form1 и показывает форму Form3 (игра)
TForm1.BitBtn2Click	Unit1	BitBtn2	Скрывает форму Form1 и показывает форму Form4 (настройки)

Продолжение таблицы 1

TForm1.BitBtn3Click	Unit1	BitBtn3	Закрывает MediaPlayer1 и завершает приложение
TForm1.BitBtn4Click	Unit1	BitBtn4	Скрывает форму Form1 и показывает форму Form5 (рекорды)
TForm1.FormCreate	Unit1	Form1	Инициализирует шрифт и музыку при создании формы
TForm1.FormShow	Unit1	Form1	Показ формы Form2 в модальном режиме при показе формы Form1
TForm1.MediaPlayer1Notify	Unit1	MediaPlayer1	Обрабатывает события уведомлений MediaPlayer1 для заикливания музыки
TForm1.N2Click	Unit1	N2	Закрывает приложение
TForm1.N4Click	Unit1	N4	Показывает сообщение об авторе
TForm1.N5Click	Unit1	N5	Показывает сообщение о версии
TForm1.SetVolume	Unit1	MediaPlayer1 (и ScrollBar1 из Unit4)	Устанавливает уровень громкости для MediaPlayer1
TForm4.BitBtn1Click	Unit4	BitBtn1	Скрывает форму Form4 и показывает форму Form1 (главная форма)
TForm4.CheckBox1Click	Unit4	CheckBox1	Управляет воспроизведением музыки (пауза/проигрывание)

Продолжение таблицы 1

TForm4.N2Click	Unit4	N2	Закрывает приложение
TForm4.N4Click	Unit4	N4	Показывает сообщение об авторе
TForm4.N5Click	Unit4	N5	Показывает сообщение о версии
TForm4.ScrollBar1Change	Unit4	ScrollBar1	Изменяет уровень громкости музыки
TForm4.CreateParams	Unit4	Form4	Обеспечивает, чтобы значок на панели задач не исчезал
TForm5.BitBtn1Click	Unit5	BitBtn1	Очищает файл рекордов и Memo1
TForm5.BitBtn2Click	Unit5	BitBtn2	Скрывает форму Form5 и показывает форму Form1 (главная форма)
TForm5.FormShow	Unit5	Form5	Загружает и сортирует записи при показе формы
TForm5.LoadRecords	Unit5	Form5	Загружает записи из файла и отображает их в Memo1
TForm5.N2Click	Unit5	N2	Закрывает приложение
TForm5.N4Click	Unit5	N4	Показывает сообщение об авторе
TForm5.N5Click	Unit5	N5	Показывает сообщение о версии
TForm5.SortRecords	Unit5	Form5	Сортирует записи в Memo1 по убыванию
TForm2.Timer1Timer	Unit2	Timer1	Обновляет позицию ProgressBar1, закрывает форму Form2 и показывает форму Form1 после завершения

Продолжение таблицы 1

TForm3.BitBtn1Click	Unit3	BitBtn1	Останавливает таймер, скрывает форму Form3 и показывает форму Form1
TForm3.BitBtn2Click	Unit3	BitBtn2	Закрывает медиаплеер, останавливает таймер и закрывает форму
TForm3.BitBtn3Click	Unit3	BitBtn3	Инициализирует игру, генерирует препятствия, перемещает змейку
TForm3.Timer2Timer	Unit3	Timer2	Перемещает змейку, проверяет столкновения и еду, перерисовывает поле
TForm3.GenerateObstacles	Unit3	Form3	Генерирует препятствия на игровом поле
TForm3.DrawGrid1DrawCell	Unit3	DrawGrid1	Отрисовывает клетки игрового поля, змейку, еду и препятствия
TForm3.InitializeGame	Unit3	Form3	Инициализирует параметры игры, включая позицию змейки, направление, еду и препятствия
TForm3.FormCreate	Unit3	Form3	Устанавливает начальные параметры формы, включая размер поля и интервал таймера

Продолжение таблицы 1

TForm3.FormKeyDown	Unit3	Form3	Обрабатывает нажатия клавиш для управления направлением змейки
TForm3.MoveSnake	Unit3	Form3	Перемещает змейку в текущем направлении
TForm3.N2Click	Unit3	N2	Закрывает приложение
TForm3.N4Click	Unit3	N4	Показывает сообщение об авторе
TForm3.N5Click	Unit3	N5	Показывает сообщение о версии
TForm3.CheckCollision	Unit3	Form3	Проверяет столкновения змейки с границами, собой и препятствиями
TForm3.UpdateScore	Unit3	Label1	Обновляет отображение текущего счета
TForm3.CheckFood	Unit3	Form3	Проверяет, съела ли змейка еду, обновляет счет и увеличивает длину змейки
TForm3.SaveRecord	Unit3	Form3	Сохраняет рекорд в файл
TForm3.GenerateFood	Unit3	Form3	Генерирует новую еду на игровом поле
TForm3.CreateParams	Unit3	Form3	Обеспечивает, чтобы значок на панели задач не исчезал
TForm3.FormShow	Unit3	Form3	Перерисовывает игровое поле при показе формы

Продолжение таблицы 1

TForm2.Timer1Timer	Unit2	Timer1	Когда прогресс достигает максимального значения, Form2 закрывается, и открывается Form1.
--------------------	-------	--------	--

3.1.2 Описание использованных компонентов

Описание использованных для разработки приложения компонентов приводится в таблице 2.

Таблица 2 – Использованные компоненты

Компонент	На какой форме расположен	Назначение
BitBtn1	TForm1	Кнопка для перехода на Form3
BitBtn2	TForm1	Кнопка для перехода на Form4
BitBtn3	TForm1, TForm3	Кнопка для выхода из приложения
BitBtn4	TForm1	Кнопка для перехода на Form5
Image1	TForm1	Изображение
Image2	TForm1, TForm3	Изображение
Image3	TForm1	Изображение
MainMenu1	TForm1	Основное меню
MediaPlayer1	TForm1	Компонент для проигрывания музыки
N1, N2, N3, N4, N5	TForm1, TForm3, TForm5	Пункты меню
ScrollBar1	TForm4	Полоса прокрутки
Timer2	TForm2, TForm3	Таймер
DrawGrid1	TForm3	Сетка для рисования
Label1	TForm3	Надпись
CheckBox1	TForm4	Флажок для управления воспроизведением музыки

Продолжение таблицы 2

RadioButton1, RadioButton2, RadioButton3	TForm4	Радиокнопки для выбора скорости змейки
RadioButton4, RadioButton5, RadioButton6	TForm4	Радиокнопки для выбора цвета змейки
RadioButton7, RadioButton8, RadioButton9	TForm4	Радиокнопки для выбора границ змейки
Edit1	TForm4	Поле для ввода текста
Memo1	TForm5	Поле для отображения записей

3.2 Спецификация программы

Точное название проекта и его состав приводится в Таблице 3.

Имя файла	Назначение
Змейка.exe	Исполняемый файл проекта
Змейка.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
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 Unit1
Unit2.dfm	Форма Unit2
Unit3.dfm	Форма Unit3
Unit4.dfm	Форма Unit4
Unit5.dfm	Форма Unit5
Icon-for-delphi.ico	Иконка для исполняемого файла
Выход bitbtn.bmp	Изображение для кнопки bitbtn
Играть bitbtn.bmp	Изображение для кнопки bitbtn
Настройка bitbtn.bmp	Изображение для кнопки bitbtn

Продолжение таблицы 3

Опции bitbtn.bmp	Изображение для кнопки bitbtn
Отмена bitbtn.bmp	Изображение для кнопки bitbtn
Очистить bitbtn.bmp	Изображение для кнопки bitbtn
Рекорд bittbn.bmp	Изображение для кнопки bitbtn
Бордик рекорды.png	Изображение для экрана в Form5
Логотип.png	Изображение для загрузочного экрана Form2
Меню.png	Изображение для экрана в Form1
Опции.png	Изображение для экрана в Form4
Очки.png	Изображение для экрана в Form3
Табло.png	Изображение для очков в Form3
Фон для загрузки.png	Изображение для загрузочного экрана в Form2
Фон.png	Изображение для экрана в Form1, Form3, Form4, Form5
terrariya-den.mp3	Саундтрек для игры
snake_record.txt	Файл для записи рекордов
press-start-k.ttf	Шрифт для отображения очков

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
1	Проверка входа пользователя в систему	Открытие главного окна	Открытие главного окна	Выполнено
2	Проверка кнопки «Играть»	Открытие формы с игрой	Открытие формы с игрой	Выполнено
3	Проверка кнопки в виде крестика	Возвращение с любой формы в главное окно	Возвращение с любой формы в главное окно	Выполнено
4	Проверка кнопки «Опции»	Открытие формы с опциями	Открытие формы с опциями	Выполнено
5	Проверка кнопки «Отключить звук» в форме с опциями	Отключение музыки	Отключение музыки	Выполнено
6	Проверка переключателей «Цвет змейки» в форме с опциями	Изменение цвет змейки	Изменение цвет змейки	Выполнено
7	Проверка переключателей «Скорость змейки» в форме с опциями	Изменение скорости змейки	Изменение скорости змейки	Выполнено
8	Проверка переключателей «Препятствий на поле» в форме с опциями	Изменение количества препятствий на игровом поле	Изменение количества препятствий на игровом поле	Выполнено

Продолжение таблицы 4

9	Проверка текстового поля «Имя игрока» в форме с опциями	Изменение никнейма для записи рекордов	Изменение никнейма для записи рекордов	Выполнено
10	Проверка scrollbar «Громкость» в форме с опциями	Изменение громкости саундтрека	Изменение громкости саундтрека	Выполнено
11	Проверка кнопки «Рекорд»	Открытие формы с рекордами	Открытие формы с рекордами	Выполнено
12	Проверка кнопки «Очистить»	Очищение текстового файла с рекордами	Очищение текстового файла с рекордами	Выполнено
13	Проверка кнопки «Выход»	Выход из программы	Выход из программы	Выполнено

При разработке программного продукта было решено множество проблем, например, не осуществлялся переход к следующему изображению по нажатию стрелки или открывался доступ к последнему квесту, когда был пройден только первый, поэтому в будущем пользователь не столкнется с данными проблемами.

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

5 Руководство пользователя

5.1 Общие сведения о программном продукте

Игровое приложение "Змейка" создано для любителей классических аркадных игр. Оно позволяет пользователям насладиться увлекательным игровым процессом, управляя змейкой и собирая пищу, чтобы увеличивать свой счет.

Для корректной работы игрового приложения "Змейка" необходимы следующие минимальные технические характеристики компьютера:

Минимальные системные требования:

- Процессор: 1.8 GHz;
- ОЗУ: 1 GB;
- От 50 MB доступного места на жестком диске;
- операционная система Windows 10.

Рекомендуемые системные требования:

- Двухъядерный процессор Intel или аналогичный AMD;
- ОЗУ: 2 GB;
- От 50 MB доступного места на жестком диске;
- операционная система Windows 10.

Игровое приложение "Змейка" оптимизировано для работы на различных конфигурациях компьютеров. Тестирование показало, что программа работает стабильно и быстро даже на устройствах с более низкими характеристиками. Оптимальная производительность достигается на компьютерах с аналогичными или вышеуказанными характеристиками.

Рекомендуется использовать игровое приложение "Змейка" на компьютерах, удовлетворяющих указанным техническим требованиям, для получения наилучшего игрового опыта. Программа легко запускается и функционирует на большинстве современных персональных компьютеров, обеспечивая приятное времяпрепровождение пользователям всех возрастных категорий.

5.2 Инсталляция

Для установки игрового приложения "Змейка" выполните следующие шаги:

1 Загрузка и установка:

- Скачайте установочный файл игры "Змейка" с официального сайта разработчика или другого авторизованного источника;
- Запустите загруженный установочный файл.

2 Установка приложения:

					КП 2-40 01 01.33.41.06.24 ПЗ	Лист
Изм.	Лист	№ докум.	Подпись	Дата		24

- Следуйте инструкциям мастера установки;
- Выберите путь для установки, если это необходимо, и завершите установку.

Запуск игры:

- После завершения установки игровое приложение "Змейка" будет доступно в меню "Пуск" или на рабочем столе компьютера;
- Запустите приложение для начала игры.

Эти простые шаги позволят быстро и легко установить игру "Змейка" на вашем персональном компьютере.

5.3 Выполнение программы

5.3.1 Запуск программы

Для запуска игрового приложения "Змейка" выполните следующие действия:

1 Настройка компьютера:

- Убедитесь, что ваш персональный компьютер соответствует минимальным системным требованиям игры "Змейка".

2 Начало запуска:

- Найдите ярлык игры "Змейка" на рабочем столе или в меню "Пуск" вашей операционной системы.

3 Запуск приложения:

- Дважды щелкните по ярлыку игры "Змейка" левой кнопкой мыши.

4 Ожидание загрузки:

- Подождите, пока игровое приложение "Змейка" загрузится. В это время может отобразиться заставка или начальный экран игры.

5 Готовность к игре:

- После загрузки приложение "Змейка" будет готово к использованию. Вы можете начать игру, следуя инструкциям на экране.

Эти шаги помогут вам без проблем запустить игровое приложение "Змейка" на вашем компьютере и насладиться игровым процессом.

					КП 2-40 01 01.33.41.06.24 ПЗ	Лист
Изм.	Лист	№ докум.	Подпись	Дата		25

5.3.2 Инструкции по работе с программой

После запуска приложения на экране нас встречает главное меню. (Рисунок 8).

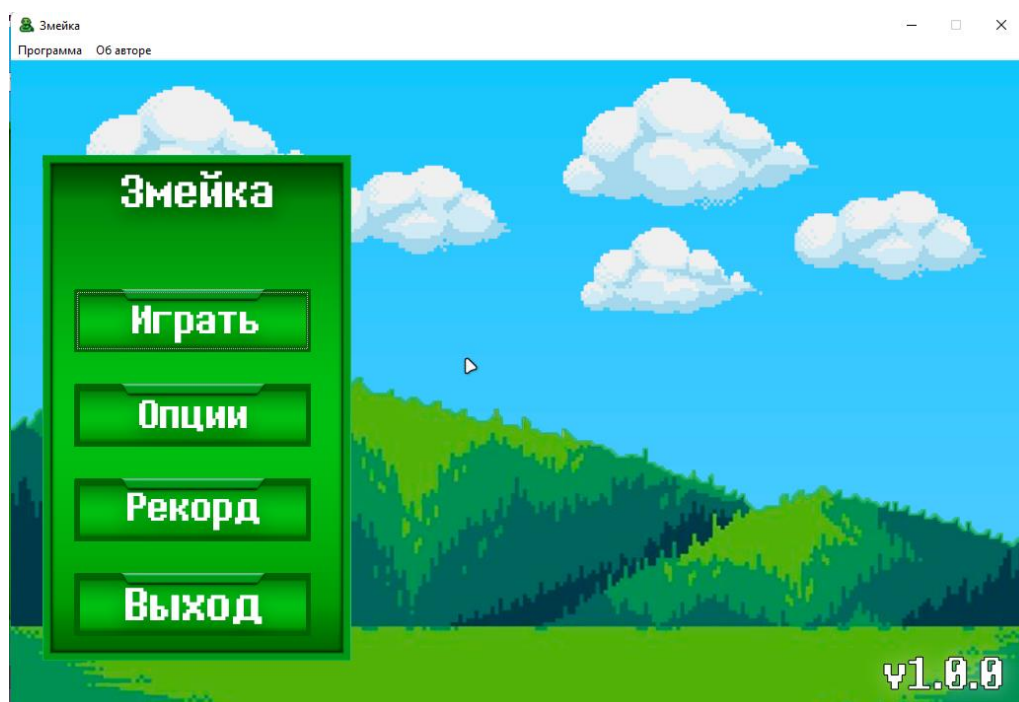


Рисунок 8 – Главное меню

При нажатии на кнопку «Играть» будет осуществлен переход на форму с игровым полем змейки (Рисунок 9), где по нажатию на «Играть» можно будет начать игру.



Рисунок 9 – Окно с игрой

По нажатию кнопки в виде крестика будет осуществлен переход в главное меню. При нажатии на кнопку «Опции» нас перекинет на окно с функциональностью настройки (Рисунок 10), в котором можно будет настроить цвет, скорость змейки, а также количество препятствий на поле и изменение имя игрока.

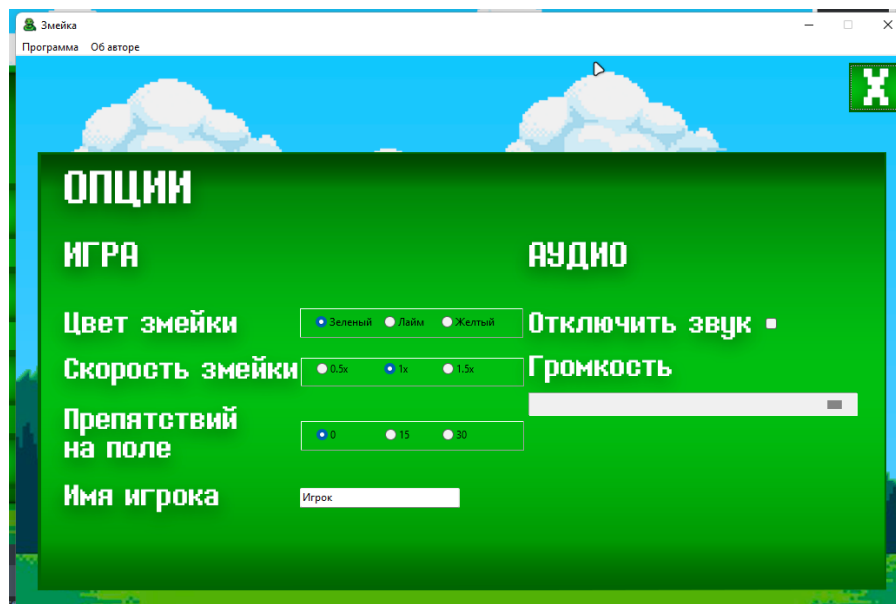


Рисунок 10 – Окно с функциональностью настройки самого приложения

По нажатию кнопки в виде крестика будет осуществлен переходе в главное меню. При нажатии на кнопку «Рекорд» нас перекинет на окно с текстовым файлом (Рисунок 11), в котором будет показано кто играл и сколько данный игрок набрал очков.

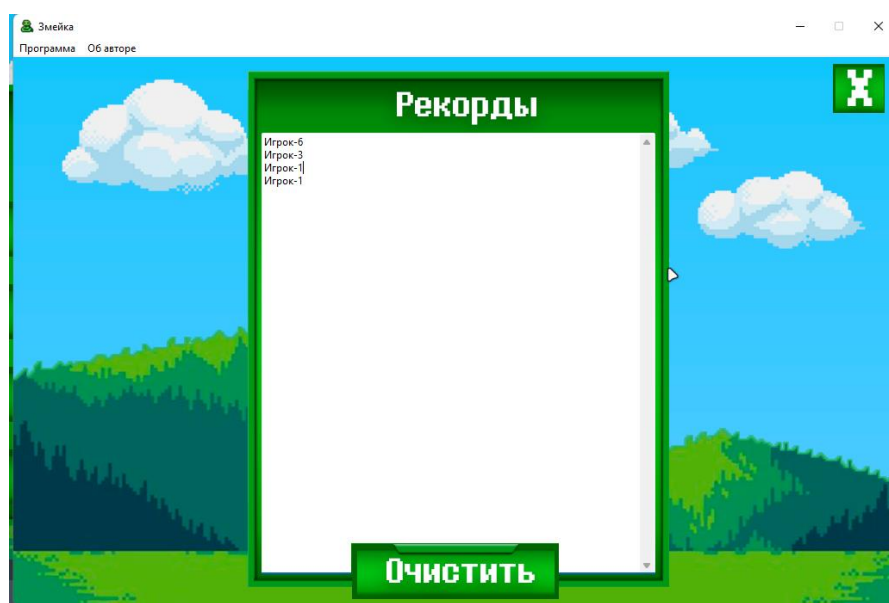


Рисунок 11 – Окно с рекордами

Завершить работу с программой можно несколькими способами:

- 1 Нужно нажать на крестик в углу формы;
- 2 Также можно закрыть игровое приложение нажав пункт Программа – Выход или нажать на кнопку главного меню «Выход». (Рисунок 12).

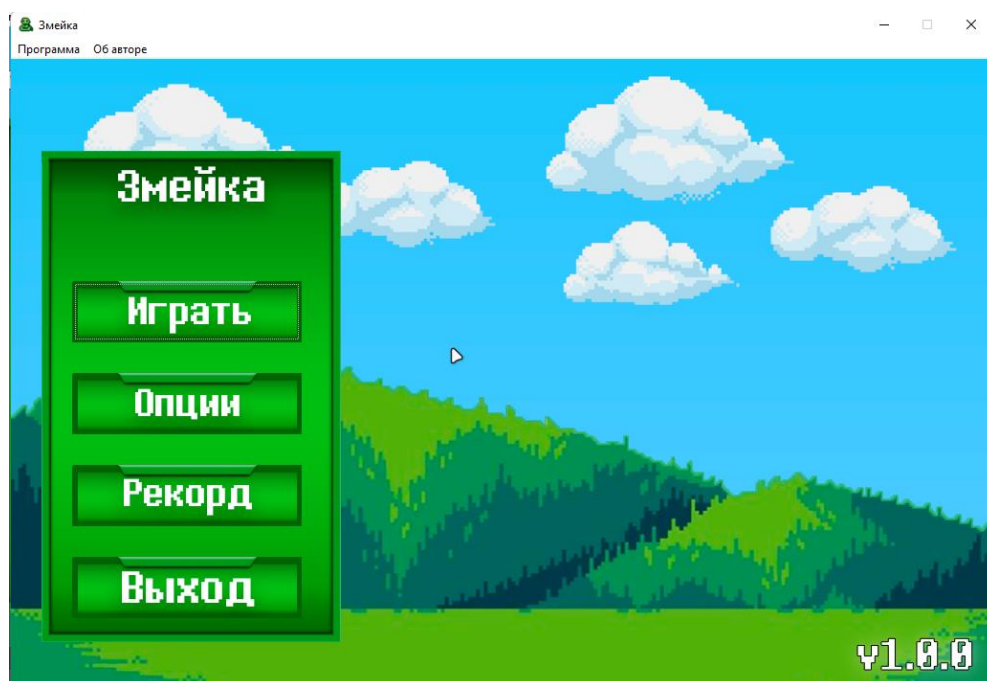


Рисунок 12 – Выход из приложения

5.3.3 Использование системы справочной информации

Справочную систему можно запустить с помощью пункта Программа – Справка (Рисунок 13).



Рисунок 13 – Справочная система

Заключение

Целью данного проекта было разработать игровое приложение "Змейка", которое представляет собой классическую аркаду с элементами логики и управления. Игра должна была быть интуитивно понятной для пользователя, обладать стабильной производительностью и быть доступной для запуска на различных платформах.

Для реализации поставленной задачи была использована среда разработки Delphi 11 с языком программирования Pascal. Основными компонентами приложения стали графическая библиотека VCL для создания интерфейса пользователя, компоненты для обработки пользовательского ввода и генерации игровых объектов.

Игра "Змейка" была создана с учетом классических правил игрового процесса: управление змейкой с помощью клавиш, генерация еды на поле, отслеживание коллизий и подсчет очков. Также были реализованы звуковые эффекты и музыкальное сопровождение с помощью компонента TMediaPlayer.

Для разработки приложения использовались следующие методы и средства:

- Язык программирования Pascal;
- Среда разработки Delphi 11;
- Графическая библиотека VCL;
- Компоненты для работы с звуком (TMediaPlayer);
- Алгоритмы генерации случайных чисел для размещения еды на поле;
- Оптимизированный код для повышения производительности;
- Степень соответствия проектных решений заданию.

Проект полностью соответствует поставленным задачам. Были реализованы все основные функции игры "Змейка", включая управление, генерацию еды, отслеживание столкновений и подсчет очков. Игра обладает интуитивно понятным интерфейсом, стабильной производительностью и может быть запущена на различных компьютерах.

Несоответствий поставленным задачам не выявлено. Все функциональные требования были выполнены в полном объеме, а производительность приложения оказалась на высоком уровне благодаря оптимизации кода.

В процессе разработки был использован нетрадиционный подход к реализации алгоритма генерации случайных позиций еды на игровом поле с использованием алгоритмов для генерации псевдослучайных чисел. Это позволило

					КП 2-40 01 01.33.41.06.24 ПЗ	Лист
Изм.	Лист	№ докум.	Подпись	Дата		29

улучшить алгоритмическую сложность и повысить эффективность работы приложения.

Игровое приложение "Змейка" остается модульным и расширяемым. Возможности модификации включают добавление новых функций, улучшение графики, внедрение дополнительных уровней сложности и интеграцию с новыми технологиями. Код приложения был структурирован таким образом, чтобы облегчить его дальнейшее развитие и поддержку.

Разработка игрового приложения "Змейка" была успешно завершена, все поставленные цели достигнуты. Проект продемонстрировал эффективное использование современных технологий разработки программного обеспечения и приложений, а также позволил расширить знания и навыки в области создания игровых приложений.

					КП 2-40 01 01.33.41.06.24 ПЗ	Лист
Изм.	Лист	№ докум.	Подпись	Дата		30

Список используемых источников

1 Delphi Форум программистов [Электронный ресурс]/ Киберфорум (cyberforum.ru). – Режим доступа: <https://www.cyberforum.ru/delphi> – Дата доступа: 18.06.2024;

2 Delphi Клуб программистов [Электронный ресурс]/ Клуб программистов (programmersforum.ru). – Режим доступа: <https://www.programmersforum.ru> – Дата доступа: 15.06.2024;

3 Delphi VINGRAD [Электронный ресурс]/ Vingrad (forum.vingrad.ru). – Режим доступа: <https://forum.vingrad.ru> – Дата доступа: 10.06.2024;

4 Delphi SNK Software [Электронный ресурс]/ SNK SOFTWARE (snkey.net). – Режим доступа: <http://www.snkey.net/forum/index.php> – Дата доступа: 13.06.2024.

					КП 2-40 01 01.33.41.06.24 ПЗ	Лист
Изм.	Лист	№ докум.	Подпись	Дата		31