Министерство просвещения Приднестровской Молдавской Республики

Государственное образовательное учреждение среднего профессионального образования «Тираспольский техникум информатики и права»

ИНДИВИДУАЛЬНЫЙ ПРОЕКТ

«Разработка приложения для просмотра рисунков»

по учебной дисциплине «Информатика»

по специальности 09.02.07 «Информационные системы и программирование»

Выполнил М. А. Давискиба

обучающийся I курса

специальность 09.02.07 «Информационные системы и программирование»

Руководитель Шандригоз Наталья Николаевна

Преподаватель информатики высшей квалификационной категории

Допущен к защите

«\_\_\_»\_\_\_\_\_\_\_202\_\_ г.

Оценка \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

Тирасполь 2024

**СОДЕРЖАНИЕ**

|  |  |
| --- | --- |
| ВВЕДЕНИЕ | 3 |
| 1. ТЕОРЕТИЧЕСКИЕ АСПЕКТЫ ЯЗЫКА ПРОГРАММИРОВАНИЯ C# | 4 |
| 1.1. Языки программирования и какие они бывают   * 1. Язык программирования C# и его возможности | 5 |
| 1.3. Приложение для просмотра рисунков. Применение и использование. | 7 |
| 1. ПРАКТИЧЕСКАЯ РЕАЛИЗАЦИЯ ИНДИВИДУАЛЬНОГО ПРОЕКТА | 8 |
| * 1. Постановка задачи   2. Реализация проекта | 10 |
| * 1. Разработка кода для приложения | 11 |
| * 1. Тестирование приложения | 12 |
| ЗАКЛЮЧЕНИЕ | 13 |
| СПИСОК ИНФОРМАЦИОННЫХ ИСТОЧНИКОВ | 14 |
|  |  |

**ВВЕДЕНИЕ**

Актуальность темы проектной работы состоит в практическом применении Windows Forms для просмотра рисунков.

Теоретическая значимость индивудального проекта заключается в разработке программного кода для просмотра изображений в Windows Forms с использование языка C#.

Практическая значимост работы состоит в следующем:

* изучить теоретические материалы по разработке в Windows Forms ;
* разработать приложение для просмотра рисунков;

Цель проектной работы – состоит в исследовании путем анализа теоретического материала и практической реализации с использованием программного обеспечения Windows Forms

Задачи исследования:

1. Определить каким функциями можно воспользоваться для создания приложения Windows Forms

2. Раскрыть основные инструменты среды разработки для создания приложения в Windows Forms

3. Разработать прикладное приложение для просмотров рисунков

4. Исследовать источники информации по разработки Windows Forms и дать рекомендации по использованию приложения

Предмет исследования – Windows Forms приложение

Объект исследования - приложение

Период исследования – март 2024

1. **ТЕОРЕТИЧЕСКИЕ АСПЕКТЫ ЯЗЫКА ПРОГРАММИРОВАНИЯ C#**
   1. **Язык программирования и виды языков.**

Языки программирования — это формальные языки для создания компьютерных программ. Язык программирования определяет набор лексических , синтаксических и семантических правил, определяющих внешний вид программы и действия, которые выполнит исполнитель под её управлением.

Со времени создания первых программированных машин человечество придумало более восьми тысяч языков программирования .Каждый год их число увеличивается. Некоторыми языками умеет пользоваться только небольшое число их собственных разработчиков, другие становятся известны миллионам людей.

Языки бывают разными, и на то есть 3 причины: Исторические, Коммерческие, Практические.

Историческая причина разделения языков: «я художник, я так вижу»

Коммерческая причина разделения языков – в том, что крупные компании заинтересованы в развитии «своих» Языков программирования

Практическая причина состоит в том, что некоторые языки заточены под ту или иную работу: C/C++ - незаменимы для разработки операционных систем и высоконагруженных приложений , Python отлично справляется с серверной частью web-разработки, Java вообще старается покрыть все сферы программирования.

Существуют основные разделения языков программирования:

По уровню: Когда вы пишете print("Hello, world!") в Python, на самом деле вы даете процессору команды вроде «Перемести из ячейки памяти в регистр процессора значение…». Компилятор переводит текст с человеческого языка на язык процессора.  
Существует два деления:

Низкоуровневые языки: Ближе к машинному коду и напрямую управляют оборудованием. Пример: Ассемблер.

Высокоуровневые языки: Высокоуровневые языки: Ближе к человеческому языку и абстрагируют детали работы оборудования. Пример: Python, Java.

По парадигме: Парадигма – это набор устоявшихся практик. Существует 3 основные парадигмы:

Императивная: Основаны на последовательных инструкциях. Пример: C.

Объектно-ориентированная: Основаны на концепции объектов, которые содержат данные и методы. Пример: Java, C++.

Функциональная: Основаны на математических функциях и избегают изменения состояния. Пример: Haskell, Lisp.

По типизации: Бывает строгой и нестрогой.

Строгая: Требует явного объявления типов переменных. Пример: Java, C#.

Нестрогая: Типы могут определяться автоматически. Пример: Python, JavaScript.

По сферам применения:

Языки для веб-разработки: Пример: JavaScript, PHP.

Языки для разработки игр: Пример: C++, C# (с использованием Unity).

Языки для разработки мобильных приложений: Пример: Swift (iOS), Kotlin (Android).

Языки для научных вычислений и анализа данных: Пример: Python, R.

* 1. **Язык программирования C# и его возможности**

На сегодняшний день язык программирования C# один из самых мощных, быстро развивающихся и востребованных в ИТ-отрасли. На нем пишутся разнообразные приложения: от небольших десктопных программ до крупных веб-порталов и веб-сервисов, обслуживающих миллионы пользователей ежедневно.

Первая версия C# вышла в феврале 2002 года вместе с релизом Visual Studio .NET. Текущая версия языка, C# 12, вышла 14 ноября 2023 года вместе с .NET 8.

C# является объектно-ориентированным и многое перенял у Java и C++. Он поддерживает полиморфизм, наследование, перегрузку операторов и статическую типизацию. Объектно-ориентированный подход позволяет решать задачи построения крупных, гибких, масштабируемых и расширяемых приложений.

Кроме того, C# активно используется в разработке игр на платформе Unity, что делает его популярным среди гейм-девелоперов. C# также поддерживает асинхронное программирование, что помогает создавать высокопроизводительные и отзывчивые приложения. Появление таких технологий, как LINQ (Language Integrated Query), упрощает работу с данными, делая код более выразительным и читаемым.

C# продолжает активно развиваться, и с каждой новой версией появляются все больше интересных функциональностей, таких как улучшенная работа с функциями высшего порядка, метапрограммирование и встроенная поддержка контейнеров и микросервисов, что делает его еще более мощным инструментом для современных разработчиков.

* 1. **. Приложение для просмотра рисунков. Применение и использование**

Приложение для просмотра рисунков - предназначены для удобного просмотра и управления изображениями.

Применение:

1. Просмотр изображений
   1. Поддержка различных форматов: JPEG, PNG, GIF, BMP, TIFF и другие.
   2. Основные функции: Масштабирование, поворот, переворот, просмотр в режиме слайд-шоу.
2. Организация и управление:
   1. Сортировка: По дате ,размеру, имени и другим параметрам.
   2. Каталогизация: Создание альбомов и коллекций.
3. Редактирование
   1. Базовые инструменты: Обрезка, изменение размера.
   2. Исправление: Поворот, выравнивание.
4. Обмен и публикация:
   1. Социальные сети и мессенджеры: Отправка изображений.
   2. Онлайн-платформы: Публикация изображений.

Использование:

1. Для домашнего использования
   1. Просмотр и хранение фотографий
   2. Создание фотоальбомов
2. Для профессионального использования
   1. Отбор и редактирование
   2. Презентация работ
3. Для Дизайнеров и Художников
   1. Просмотр и управление работами.
   2. Интеграция с графическими редакторами

Примеры популярных приложений для просмотра рисунков

1. InfanView - Легкий и быстрый редактор с базовыми инструментами

2. FastStone Image Viewer - Многофункциональное приложение для просмотра рисунков и редактирование

3. XnView - Поддержка множества форматов и мощные инструменты для органищцаии

**2. ПРАКТИЧЕСКАЯ РЕАЛИЗАЦИЯ ИНДИВИДУАЛЬНОГО ПРОЕКТА**

**2.1 Постановка задачи**

Задача: Разработка приложения для просмотра рисунков.

Цель: Создать удобное и эффективное приложение, которое позволяет пользователям легко просматривать изображения.

Основные требования:

Функции просмотра изображений:

Форматы: Поддержка JPEG, PNG, GIF, BMP, TIFF.

Просмотр: Масштабирование .

Функция редактирования изображений:

Базовые инструменты: изменение размера, корректировка цвета.

Технические требования:

1. Интерфейс: Интуитивно понятный и удобный интерфейс пользователя.

2. Совместимость: Поддержка операционных систем Windows, macOS и Linux.

3. Производительность: Высокая производительность и быстрая обработка изображений.

Критерии успеха:

1. Легкость использования приложения.

2. Поддержка всех заявленных форматов и функций.

3.Высокая производительность и стабильность приложения.

Эта постановка задачи предоставляет четкое представление о цели, требованиях, критериях успеха, необходимых для успешного выполнения проекта по разработке приложения для просмотра рисунков.

**2.2 Реализация задачи**

Данная приложение была реализована на основе ,,Учебник. Создание средства просмотра изображений приложения Windows Forms в Visual Studio” в котором была расписана пошаговая реализация приложения.

В инструкции было 3 этапа создания:

1. Создание проекта и добавление элементов пакета.
2. Добавление элементов управления в средство для просмотра изображений.
3. Добавление кода в средство просмотра изображений.

Учебник детально описал и научид:

1. Как создавать проект Windows Forms в Visual Studio
2. Добавлять макет для приложения просмотров рисунков
3. Добавление кнопок и флажков
4. Добавление диалоговых окон
5. Добавление обработчиков событий для элементов управления
6. Написание кода на C#

Для хранения и управления разработкой приложения для просмотра рисунков используется репозиторий, созданный в сервисе GitHub. Эта система контроля версий предоставляет множество возможностей для эффективной работы над проектом. На GitHub создаётся репозиторий, где хранятся все файлы проекта, что позволяет централизованно управлять кодом и обеспечивать доступ ко всем версиям.

Ссылка на репозиторий : https://github.com/DarkTimeMonster/AppScreen

* 1. **Разработка кода для приложения**

namespace WinFormsApp2

{

public partial class Form1 : Form

{

public Form1()

{

InitializeComponent();

}

private void tableLayoutPanel1\_Paint(object sender, PaintEventArgs e)

{

}

private void button1\_Click(object sender, EventArgs e)

{

//Данный код необходим для отоброжения файлов изображений

if (openFileDialog1.ShowDialog() == DialogResult.OK)

{

pictureBox1.Load(openFileDialog1.FileName);

}

}

private void button2\_Click(object sender, EventArgs e)

{

// Данный код необходим для очистки рисунка , который вы открыли в диалоговом окне

pictureBox1.Image = null;

}

private void openFileDialog1\_FileOk(object sender, System.ComponentModel.CancelEventArgs e)

{

}

private void Form1\_Load(object sender, EventArgs e)

{

}

private void backgroundButton\_Click(object sender, EventArgs e)

{

// Данный код изменяет задний фон диалогового окна

if (colorDialog1.ShowDialog() == DialogResult.OK)

pictureBox1.BackColor = colorDialog1.Color;

}

private void closeButton\_Click(object sender, EventArgs e)

{

// Данный код закрывает приложение

this.Close();

}

private void checkBox1\_CheckedChanged(object sender, EventArgs e)

{

// Данный код необходим для растяжения рисунка.

//Если пользователь ставит флажок растянуть то свойству SizeMode присваивается значение "Stretch" (растянуть)

// Если пользователь убирает флажок с растянуть то свойству SizeMode присваивается значение “Normal” (Обычный)

if (checkBox1.Checked)

pictureBox1.SizeMode = PictureBoxSizeMode.StretchImage;

else

pictureBox1.SizeMode = PictureBoxSizeMode.Normal;

}

private void pictureBox1\_Click(object sender, EventArgs e)

{

}

private void flowLayoutPanel1\_Paint(object sender, PaintEventArgs e)

{

}

}

}

* 1. **Тестирование приложения**

Рисунок без растяжения:

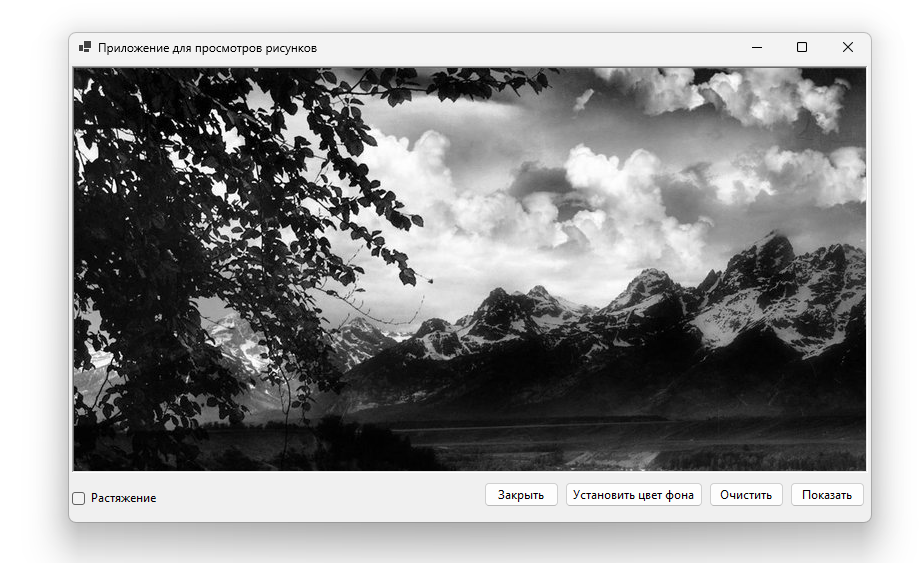
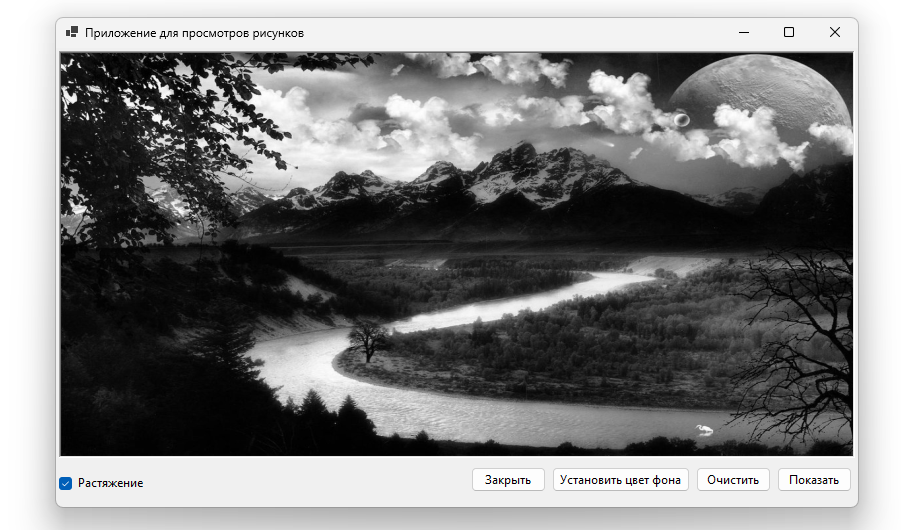
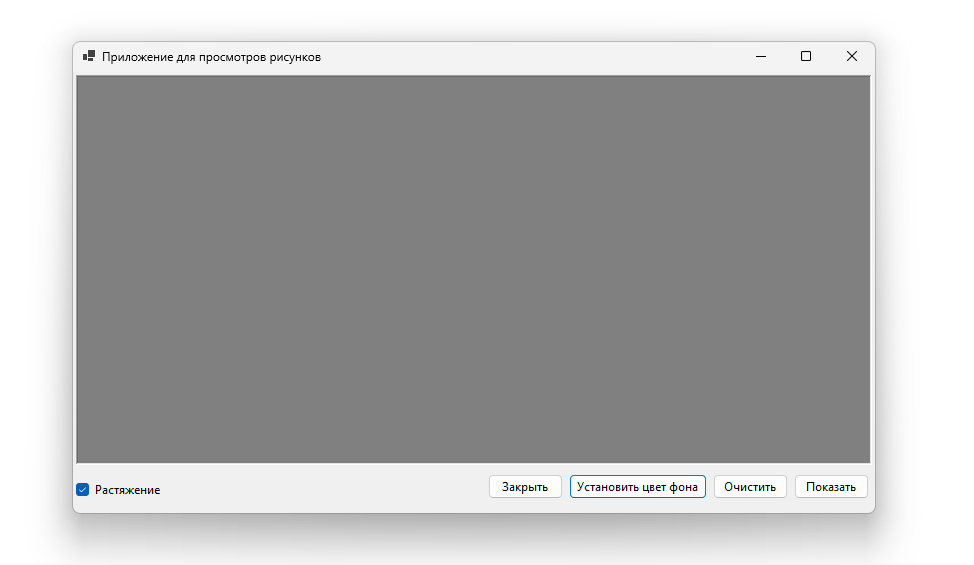
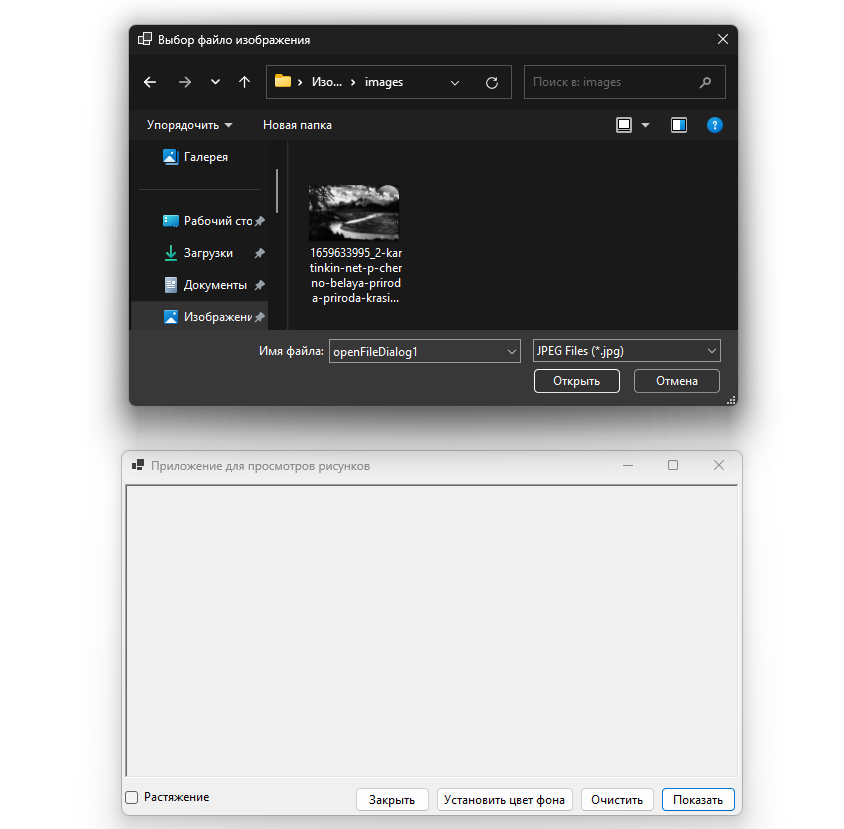


Рисунок c растяжением:



Установлен задний фон:



Открытие приложением файлов рисунков:

**ЗАКЛЮЧЕНИЕ**

В ходе выполнения проектной работы решены следующие задачи:

1 Изучена информация по WinForms разработке для разработки приложения просмотров рисунков. В процессе работы были рассмотрены ключевые этапы от создание проекта до его завершения.

1. Изучены приложения для просмотров рисунков и их виды.
2. Использование GitHub для хранения и управления проектом оказалась крайне полезной.
3. Изучены материалы ,,Учебник. Создание средства просмотра изображений приложения Windows Forms в Visual Studio “

5 Проект показал, как современные инструменты и технологии могут быть использованы для создания полезных и эффективных приложений. Работа над этим приложением предоставила ценные знания и навыки .

**СПИСОК ИНФОРМАЦИОННЫХ ИСТОЧНИКОВ**

1 \*\*Microsoft Learn - C# Documentation\* - Ссылка: [<https://learn.microsoft.com/en-us/dotnet/csharp/> ]

- Описание: Официальная документация по C# от Microsoft. Включает учебные материалы, примеры кода и справочную информацию.

2 \*\*C# Programming Guide - Microsoft Docs\*\* - Ссылка [<https://learn.microsoft.com/en-us/dotnet/csharp/programming-guide/> ]

- Описание: Полный гид по программированию на C#, охватывающий все основные аспекты языка.

3 \*\*C# Tutorials - Microsoft Docs\*\* - Ссылка: [<https://learn.microsoft.com/en-us/dotnet/csharp/tutorials/> ]

- Описание: Набор пошаговых руководств и примеров для начинающих и продвинутых разработчиков.

4 \*\*Windows Forms Documentation - Microsoft Learn\* Ссылка: [ [<https://learn.microsoft.com/en-us/dotnet/desktop/winforms/>]

- Описание: Официальная документация по Windows Forms от Microsoft. Включает руководство по использованию Windows Forms для создания приложений с графическим интерфейсом.

5 \*\*Creating a Windows Forms App - Microsoft Docs\*\*Ссылка: [<https://learn.microsoft.com/en-us/visualstudio/ide/create-csharp-winform-visual-studio?view=vs-2022>]

- Описание: Руководство по созданию приложений Windows Forms в Visual Studio.

6 \*\*Windows Forms Tutorials - CodeProject\*\* - Ссылка: [<https://www.codeproject.com/KB/tags/windows-forms.aspx>]

- Описание: Коллекция статей и руководств по Windows Forms от сообщества разработчиков на CodeProject.

7 \*\*GeeksforGeeks - Programming Languages\*\* - Ссылка: [<https://www.geeksforgeeks.org/category/programming-language/> ]

- Описание: Разнообразные статьи и учебные материалы по различным языкам программирования, включая C++, Python, Java и другие.

8 \*\*Codecademy - Learn to Code\* - Ссылка: [<https://www.codecademy.com/learn/paths/computer-science> ] - Описание: Онлайн-курсы по программированию, охватывающие множество языков программирования и концепций.

9 \*\*Coursera - Programming Languages Courses\*\*- Ссылка: [<https://www.coursera.org/courses?query=programming%20languages> ] - Описание: Курсы по программированию от ведущих университетов и организаций. Курсы включают обучение различным языкам программирования.

10 \*\*Pluralsight - C# Coureses\* - Ссылка: [[pluralsight.com/paths/cshap](https://learn.microsoft.com/en-us/dotnet/csharp/) ]

- Описание: Курсы по C# от начинающих до продвинутых уровней.