Министерство науки и высшего образования Российской Федерации

ФГАОУ ВО «УрФУ имени первого Президента России Б.Н. Ельцина»

Нижнетагильский технологический институт (филиал)

Кафедра информационных технологий

Оценка работы\_\_\_\_\_\_\_\_\_\_\_\_\_\_

Преподаватель Познякова К.А

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

«\_\_\_»\_\_\_\_\_\_\_\_\_\_\_\_20\_\_\_г.

Пояснительная записка

Редактор графических изображений

.

Студент Сергеичев А.В

Специальность (направление подготовки) ИСиТ

Группа ТВ-473902

«\_\_\_\_» \_\_\_\_\_\_\_\_\_\_\_\_2021

Нижний Тагил

2021

Содержание

Введение

1. Системотехническая часть

1.1 Описание и анализ предметной области

1.1.1 Основные понятия предметной области

1.1.2 Классификация графических редакторов

1.1.3 Описание систем-аналогов

1.1.4 Анализ предметной области

1.2 Постановка задачи

1.3 Структурная схема системы

1.4 Разработка UML-проекта

1.4.1 Диаграмма вариантов использования

1.4.2 Диаграмма состояний

1.4.3 Диаграмма классов

1.4.4 Диаграмма деятельности

1.5 Разработка прототипа интерфейса пользователя

1.6 Выбор и обоснование комплекса программных средств

1.6.1 Выбор языка программирования и среды разработки

1.6.2 Выбор операционной системы

1.6.3 Выбор языка программирования

1.6.4 Выбор среды разработки

2. Конструкторско-технологическая часть

2.1 Разработка и описание интерфейса пользователя

2.2 Разработка и описание пользовательского меню

2.3 Описание тестового примера

Заключение

Список использованных источников

Введение

Представление данных на мониторе компьютера в графическом виде впервые было реализовано в середине 50-х годов для больших ЭВМ, применявшихся в научных и военных исследованиях. С тех пор графический способ отображения данных стал неотъемлемой частью подавляющего числа компьютерных систем, в особенности персональных.

Существует специальная область информатики, изучающая методы и средства создания и обработки изображений с помощью программно-аппаратных вычислительных комплексов – компьютерная графика. Она охватывает все виды и формы представления изображений, доступных для восприятия человеком либо на экране монитора, либо в виде копии на внешнем носителе (бумага, киноплёнка, ткань и прочее). Без компьютерной графики невозможно представить себе не только компьютерный, но и обычный, вполне материальный мир. На сегодняшний день компьютеры и компьютерная графика неотъемлемая часть жизни современного общества. Для примера назовём медицину (компьютерная томография), научные исследования (визуализация строения вещества, векторных полей и других данных), моделирование тканей и одежды, опытно-конструкторские разработки, рекламные щиты, цветные журналы, спецэффекты в фильмах – всё это в той или иной мере имеет отношение к компьютерной графике. Поэтому созданы программы для создания и редактирования изображений, то есть графические редакторы.

Графический редактор – это программа, предназначенная для автоматизации процессов построения на экране дисплея графических изображений. Предоставляет возможности рисования линий, кривых, раскраски областей экрана, создания надписей различными шрифтами и т.д.

Большинство редакторов позволяют обрабатывать изображения, полученные с помощью сканеров, а также выводить картинки в таком виде, чтобы они могли быть включены в документ, подготовленный с помощью текстового редактора.

Некоторые редакторы позволяют получать изображения трёхмерных объектов, их сечений, разворотов, каркасных моделей и т.п.

Цель проекта – разработать графический редактор "Редактор графических изображений", который по функциям будет аналогичен редактору MS Paint.

Проектирование будет вестись по методологии UML (англ. Unified Modeling Language – унифицированный язык моделирования). UML− язык графического описания для объектного моделирования в области разработки программного обеспечения. UML является языком широкого профиля, это открытый стандарт, использующий графические обозначения для создания абстрактной модели системы, называемой UML-моделью. UML был создан для определения, визуализации, проектирования и документирования, в основном, программных систем.

1. Системотехническая часть

### 1.1 Описание и анализ предметной области

### 1.1.1 Основные понятия предметной области

Прежде чем приступить к реализации любого проекта, нужно хорошо представлять предметную область поставленной задачи.

Предметная область – совокупность основных объектов, участвующих в функционировании системы. Целью данного проекта является создание графического редактора растровых изображений.

Компьютерная графика – специальная область информатики, изучающая методы и средства создания и обработки изображений с помощью программно-аппаратных вычислительных комплексов [2]. К вычислительным комплексам относятся графические редакторы.

Графический редактор – это программа, позволяющая создавать, редактировать и просматривать изображения на экране компьютера.

Сами изображения условно подразделяют на такие классы, как картины, фотографии, чертежи. В соответствии с этим существуют графические редакторы, ориентированные на художников и дизайнеров, фотографов, инженеров.

По способу обработки изображения графические редакторы подразделяются на растровые, векторные и гибридные.

Растровое изображение – изображение, представляющее собой сетку пикселей или цветных точек (обычно прямоугольную) на мониторе, бумаге и других отображающих устройствах и материалах [2]. К основным характеристикам растрового изображения относятся: количество пикселей, глубина цвета, цветовая модель и разрешение изображения. Пример растрового изображения приведен на рисунке 1.

Векторные рисунки формируются из базовых графических объектов, для каждого из которых задаются координаты опорных точек, а также цвет, толщина и стиль линии его контура. Векторное изображение может свободно масштабироваться, не теряя при этом качества. Пример растрового изображения приведен на рисунке 1.

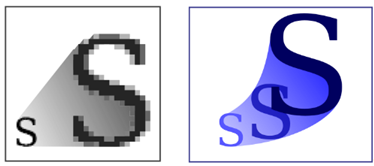


Рисунок 1 – Пример растрового и векторного изображения

### 

### 1.1.2 Классификация графических редакторов

Графические редакторы предназначены для обработки на компьютере графических изображений. По способу обработки изображения графические редакторы подразделяются на растровых и векторные.

Растровый графический редактор представляет изображение в виде набора точек – пикселей. Из растровых графических редакторов популярностью пользуются Adobe Photoshop, GIMP.

Векторные редакторы сохраняют информацию об элементах изображении в виде координат в некоей системе отсчета. Наиболее популярные векторные графические редакторы: CorelDraw, AdobeIllustratore, Inkscape.

В настоящее время наиболее распространенным и простым растровым графическим редактором является MS Paint.

### 

### 1.1.3 Описание систем-аналогов

На данный момент существует огромное множество растровых графических редакторов. Рассмотрим самые популярные из них.

Adobe Photoshop – многофункциональный графический редактор, разработанный и распространяемый фирмой Adobe Systems. Продукт выступает лидером современного рынка коммерческих инструментов, позволяющих проводить процессы редактирования растрового изображения. Сегодня работать посредством программы можно базируясь на операционных системах Windows, Mac.

Приложение обладает уникальной функцией интеграции других программных проектов, обрабатывающими мультимедиа файлы, анимацию, иное творчество. Приложение нашло широкое применение в сфере разработки компьютерных игр, создания профессиональных дисков, обеспечивает производство процессов линейного монтажа, создания четких спецэффектов, текстуры, фона телевидения, кинематографа.

Программное обеспечение обладает огромным количеством уникальных функций. Имеется широкий диапазон цветовых характеристик, обрабатываемых приложением. Есть возможность сохранять посредством единого файла одновременно несколько элементов – направляющие, каналы прозрачности, пути отправки, содержание текстовых, векторных объектов, слоев. Рабочее окно Adobe Photoshop показано на рисунке 2.

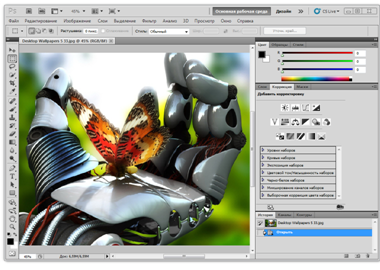


Рисунок 2 – Рабочее окно графического растрового редактора Adobe Photoshop

GIMP (англ. General Image Manipulation Program – главная программа растровых изображений), в отличие от Adobe Photoshop, является бесплатной многофункциональная программа для работы с изображениями, которую можно дополнить с помощью значительного количества бесплатных плагинов, существенно расширяющих её возможности. GIMP поддерживает большинство существующих форматов. Программа включает в себя полный комплект необходимых инструментов для обработки изображений, каждый из которых имеет множество тонких настроек. Неограниченное количество одновременно открываемых фотографий, поддержка слоев, пакетная обработка, наличие русского языка и много чего другого делают GIMP очень удобным и мощным инструментом. Рабочее окно редактора GIMP изображено на рисунке 3.

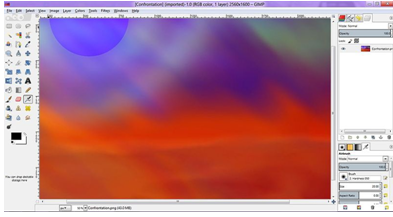


Рисунок3 – Рабочее окно графического растрового редактора GIMP

Paint – является простым и графическим редактором, входящий в состав стандартных программ Windows. Он также часто упоминается как MS Paint или Microsoft Paint. Позволяет создавать рисунки, картинки и т.д. или редактировать изображения. Кроме того, его можно использовать для сохранения графических файлов в различных форматах.

Программа может работать в цветном режиме или двухцветном черно-белом. Из-за своей простоты он быстро стал одним из самых используемых приложений в ранних версиях. Экранная форма MS Paint ОС Windows 7 и пример рисунка, созданного с его помощью, представлен на рисунке 4.

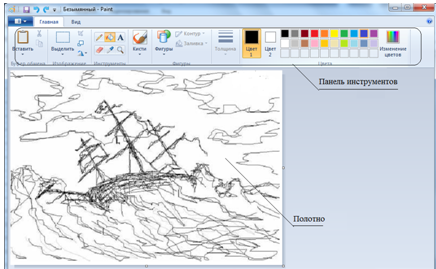
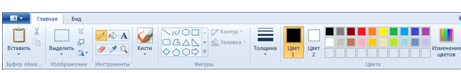


Рисунок 4 – Экранная форма MS Paint

Необходимые инструменты для создания изображения представлены на рисунке 5.

Рисунок 5 – Панель инструментов MS Paint.

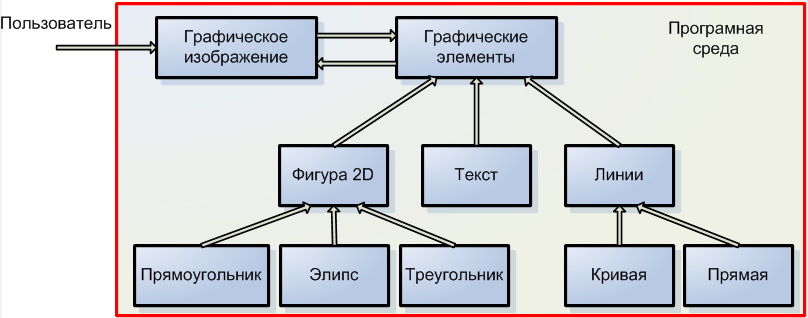
### 

### 1.1.4 Анализ предметной области

Для выделения или идентификации компонентов предметной области был использован принцип концептуализации предметной области.

В результате анализа были выделены следующие основные объекты:

* Графическое изображение – создается пользователем и состоит из различных графических элементов;
* Графические элементы (инструменты) – предназначены для создания изображения и включают в себя двумерные фигуры, текст и линии;
* Двумерные фигуры включают в себя многоугольник, прямоугольник, эллипс и закругленный прямоугольник;
* Линия – может быть прямой или кривой.



Таким образом, всем процессом создания изображения управляет пользователь, который создает графическое изображение и применяет к нему различные действия.

* создание нового изображения;
* сохраннее текущего изображения;
* открытие ранее созданного изображения;
  + редактирование изображения;
* форматирование изображения:

### 

### 1.2 Постановка задачи

В рамках проекта необходимо разработать программу "Редактор графических изображений", позволяющую производить обработку растровых изображений.

Графическое изображение изначально с помощью имеющихся в наборе инструментов должно загружаться из файла.

Разрабатываемая система будет реализовывать следующие функции:

1. сохранение графического изображения;
2. загрузка графического изображения;
3. редактирование графического изображения:

* выделение фрагмента изображения;
* вырезка изображения;
* разворот зеркально;
* разворот по вертикали;
* разворот по горизонтали;
* копирование изображения;
* вставка изображения;

1. изменение масштаба изображения;
2. форматирование изображения:

* изменение размера изображения;
* наложение цветных фильтров;
* фильтр бинаризация;
* фильтр Морфология;
* фильтр свертка и корреляция;
* фильтр края детектор;
* фильтр медианный;

1. отмена предыдущего действия.

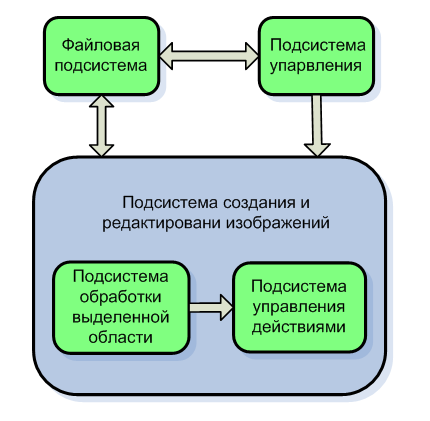
### 

### 1.3 Структурная схема системы

Система – это совокупность объектов, компонентов или элементов произвольной природы, образующих некоторую целостность. Более мелкая система – это подсистема.

Структурная схема отображает принцип работы системы или устройства в самом общем виде. На схемах изображаются все основные функциональные элементы, а также основные взаимосвязи между ними, указывающие на последовательность взаимодействия функциональных элементов в схеме.

Структурная схема системы приведена на рисунке 5, она позволяет определить основные подсистемы, а также связи между ними.



Данный проект можно разделить на подсистемы, представленные в таблице 2.

Таблица 2‑ Перечень подсистем

|  |  |  |
| --- | --- | --- |
| 1 | | 2 |
| 1. П/c управления | | Необходима для организации взаимодействия подсистем. |
|  | П/с создания и редактирования изображения | Необходима для создания и редактирования изображения |
| 2.1.П/с обработки выделенной области | Необходима для работы с выделенными фрагментами изображения. |
| 2.2.П/с управления действиями | Предназначена для реализации различных действий над изображением. |
| 1. Файловая п/с | | Предназначена для работы с файлами: загрузки уже существующего изображения, сохранения нового или измененного изображения. |

### 

### 1.4 Разработка UML-проекта

Язык UML (Unified Modeling Language) представляет собой унифицированный язык визуального моделирования, который разработан для специфицирования (создания спецификации), конструирования, визуализации и документирования компонентов программного обеспечения и бизнес-процессов.

UML позволяет также разработчикам программного обеспечения достигнуть соглашения в графических обозначениях для представления общих понятий (таких как класс, компонент, обобщение (generalization), объединение (aggregation) и поведение), и больше сконцентрироваться на проектировании и архитектуре.

### 

### 1.4.1 Диаграмма вариантов использования

Диаграмма вариантов использования описывает функциональное назначение системы. Проектируемая система представляется в виде множества сущностей или актеров, взаимодействие которых с системой отображается в виде взаимосвязанных вариантов использования.

Взаимодействие действующих лиц и вариантов использования между собой описывается с помощью отношений: ассоциации, включения, расширения, обобщения.

Диаграмма вариантов использования разрабатываемой системы представлена на рисунке 9. Разрабатываемая система содержит одного актера - пользователь. Пользователь будет иметь возможность создавать изображение, сохранять его, загружать раннее созданный, настраивать внешний вид системы, редактировать изображение в рабочем поле, поворачивать изображение, изменять цвет изображения. Также пользователь может изменять размер, начертание, цвет.

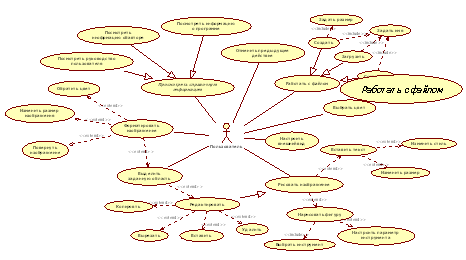


Рисунок 9– Диаграмма вариантов использования

### 1.4.2 Диаграмма состояний

Диаграмма состояний описывает процесс изменения состояний только одного класса, а точнее – одного экземпляра определенного класса, т. е. моделирует все возможные изменения в состоянии конкретного объекта. При этом изменение состояния объекта может быть вызвано внешними воздействиями со стороны других объектов или извне. Именно для описания реакции объекта на подобные внешние воздействия и используются диаграммы состояний.

Главное предназначение этой диаграммы – описать возможные последовательности состояний и переходов, которые в совокупности характеризуют поведение элемента модели в течение его жизненного цикла.

Диаграмма состояний представляет динамическое поведение сущностей, на основе спецификации их реакции на восприятие некоторых конкретных событий. Системы, которые реагируют на внешние действия от других систем или от пользователей, иногда называют реактивными. Если такие действия инициируются в произвольные случайные моменты времени, то говорят об асинхронном поведении модели.

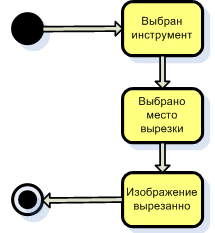
На рисунке 10 представлена диаграмма состояний графическом редактора при работе с обрезкой изображения. Пользователь заходит в меню, изображение, расположенное на главной форме. Далее необходимо выбрать на форме место вырезки изображения, затем отпустить левую кнопку мыши, в результате чего вырезанное изображение отобразится на форме.

Рисунок 10 – Диаграмма состояний при работе с текстом

### 

### 1.4.3 Диаграмма классов

Для представления абстракций объектов используется специальный определяемый программистом тип данных – класс. Существует несколько определений класса.

На рисунке 11 изображена диаграмма классов разрабатываемой автоматизированной системы. Все классы являются стандартными классами, входящими в состав Microsoft Visual Studio 2017.



Рисунок 11 – Диаграмма классов

Класс "Главная форма" облает расширенной функциональностью для форматирования, а также изменения параметров текста, сохранения и открытия файла.

1.4.4 Диаграмма деятельности

Диаграмма деятельности – диаграмма, на которой показано разложение некоторой деятельности на её составные части. Под деятельностью понимается спецификация исполняемого поведения в виде координированного последовательного и параллельного выполнения подчинённых элементов - вложенных видов деятельности и отдельных действий, соединённых между собой потоками, которые идут от выходов одного узла к входам другого.

На рисунке 12 показана диаграмма деятельности для варианта использования "Редактирование выделенной области". Для выделения области необходимо выбрать на панели инструментов соответствующий инструмент и произвести выделение части изображения, после чего выбрать соответствующий пункт контекстного меню. Пользователю предоставляется возможность вырезать, удалить, копировать, использовать многократно вставку изображения и поворот изображения.

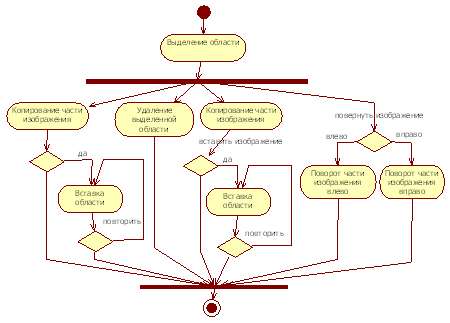


Рисунок 12 – Диаграмма деятельности для варианта использования "Редактирование выделенной области"

1.5 Разработка прототипа интерфейса пользователя

Разработка интерфейса пользователя является одним из важнейших этапов проектирования системы. Она начинается с ранних набросков диалоговых окон на этапе анализа требований. Эти наброски используются в процессе сбора требований, при разборе возможных сценариев работы системы с заказчиками, для создания прототипов и для включения документов описания прецедентов. В процессе проектирования осуществляется дальнейшая разработка окон пользовательского интерфейса в соответствии с основными возможностями презентационного ПО, а также особенностями и ограничениями выбранной программной среды.

Пользовательский интерфейс должен быть интуитивно понятным, дружественным и удобным. Программист должен поставить себя на место пользователя, который впервые видит программу, и создать такой интерфейс, который не потребует много времени на обучение, понимание и восприятие программы, а также на работу с окнами.

На рисунке 14 представлен прототип интерфейса графического редактора.

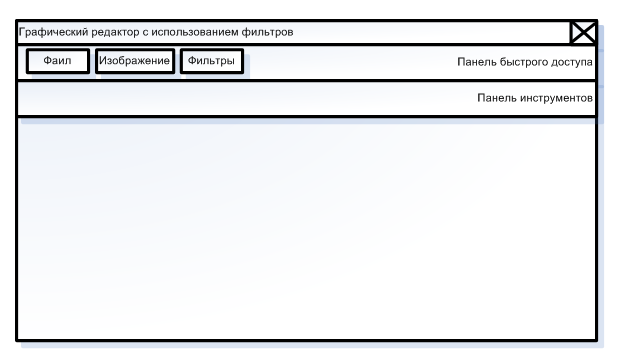


Рисунок 14 – Прототип интерфейса графического редактора

В верхней части окна редактора располагаются вкладки "Файл", "Изображение", "Фильтры". Под ними находится панель быстрого доступа к основным функциям системы – копирование, обрезать, зум и центрирование рабочего фона.

1.6 Выбор и обоснование комплекса программных средств

1.6.1 Выбор языка программирования и среды разработки

Пакет Microsoft Visual Studio.NET является интегрированной средой разработки (IDE - Integrated Development Environment), потому что с помощью одного только этого средства, не прибегая к помощи других, можно выполнить многое, а именно:

* генерировать "скелет" приложения без написания кода вручную;
* открывать проект в нескольких различных режимах представления;
* редактировать все составляющие кода как вручную, так и с помощью мастеров;
* разрабатывать визуальный интерфейс (меню и диалоговые окна);
* подключаться к внешним ресурсам - например, к базам данных;
* компилировать и линковать исходный код;
* отлаживать приложение в процессе его работы.

Среда разработки Visual Studio.NET имеет единый интерфейс для всех включенных в нее языков программирования: C#, VB.NET, C++ и J++. Язык программирования C# является полнофункциональным объектно-ориентированным языком, который поддерживает все три "столпа" объектно-ориентированного программирования: инкапсуляцию, наследование и полиморфизм. Он имеет прекрасную поддержку компонентов, надежен и устойчив благодаря использованию "сборки мусора", обработки исключений, безопасности типов. Язык программирования С++ является типизированным языком программирования общего назначения. Поддерживая разные парадигмы программирования, сочетает в себе свойства как высокоуровневых, так и низкоуровневых языков. В сравнении с его предшественником – языком C, наибольшее внимание уделено поддержке объектно-ориентированного и обобщённого программирования. Исходя из характера поставленной задачи, технических требований и наработанного инструментария нами была выбрана среда разработки Visual Studio.NET и язык программирования C#.

### 

### 1.6.2 Выбор операционной системы

Программа предназначена для работы в операционных системах (OC) Microsoft Windows XP/7 и выше. Данные ОС являются наиболее распространёнными в мире. Удобный и наглядный оконный интерфейс ОС семейства Windows предоставляет широкие возможности по реализации визуального оформления программы, а встроенные службы позволяют организовать легкий доступ к руководству пользователя.

OC Windows предоставляет программистам все необходимые средства для создания пользовательского интерфейса. Поэтому пользовательский интерфейс программ в значительной степени унифицирован, и пользователям не требуется изучать для каждой программы новые принципы организации взаимодействия с этой программой. К достоинству OC Windows можно отнести доступность всей оперативной памяти, что облегчает создание больших программ. Очередным преимуществом OC Windows является динамическое подключение библиотек – OC Windows автоматически подключает библиотеки подпрограмм во время выполнения программы. Формат библиотек и порядок вызова библиотечных подпрограмм стандартизированы, поэтому эти библиотеки могут быть созданы с помощью различных программных средств.

### 

### 1.6.3 Выбор языка программирования

Среда разработки Visual Studio.NET представляет мощные и удобные средства написания, корректировки, компиляции, отладки и запуска приложений, использующих .NET – совместимые языки. Корпорация Microsoft включила в платформу средства разработки для четырех языков: C#, VB.NET, C++ и J++. Язык программирования C# является полнофункциональным объектно-ориентированным языком, который поддерживает все три "столпа" объектно-ориентированного программирования: инкапсуляцию, наследование и полиморфизм. Он имеет прекрасную поддержку компонентов, надежен и устойчив благодаря использованию "сборки мусора", обработки исключений, безопасности типов. Исходя из характера поставленной задачи, технических требований и наработанного инструментария нами была выбрана среда разработки Visual Studio.NET и язык программирования C#.

### 

### 1.6.4 Выбор среды разработки

В качестве среды программирования была выбрана Visual Studio 2017, так как она включает в себя новые возможности, такие как наглядные средства разработки для более быстрой работы с .NET Framework 4.0 и улучшения языков, ускоряющие работу со всеми типами данных. Что самое важное, так это то, что Visual Studio 2017 предлагает усовершенствованные средства, помогающие наладить совместную деятельность в группах.

Microsoft Visual Studio 2017 также помогает индивидуальным программистам и небольшим группам, создающим любые виды ПО, ускорить разработку приложений и создание пользовательских интерфейсов с принципиально новым уровнем удобства, повысить эффективность коллективной работы. Используя большое количество новых функций в Visual Studio 2017, российские разработчики могут решать сложные бизнес-задачи при одновременном сокращении совокупной стоимости процесса разработки. А вместе с долгожданным нововведением – выпуском полностью локализованного на русский язык продукта – производительность труда профессиональных разработчиков возрастет в разы, ведь теперь весь интерфейс продукта вместе со всей документацией будут представлены на родном языке.

2. Конструкторско-технологическая часть

### 

### 2.1 Разработка и описание интерфейса пользователя

Большинство программных продуктов, особенно прикладного характера, ориентированных на конечного пользователя, работают в диалоговом режиме взаимодействия с пользователем таким образом, что ведется обмен сообщениями, влияющими на обработку данных. В диалоговом режиме под воздействием пользователя осуществляются запуск функций (методов) обработки, изменение свойств объектов, производится настройка параметров выдачи информации на печать и т.п.

Графический интерфейс пользователя является обязательным компонентом большинства современных программных продуктов, ориентированных на работу конечного пользователя. К графическому интерфейсу пользователя предъявляются высокие требования как с чисто инженерной, так и с художественной стороны разработки, при его разработке ориентируются на возможности человека. Наиболее часто графический интерфейс реализуется в интерактивном режиме работы пользователя для программных продуктов, функционирующих в OC Windows, и строится в виде системы выпадающих меню с использованием в качестве средства манипуляции мыши и клавиатуры .

Основная экранная форма графического редактора приведена на рисунке 17.

На главной форме программы располагаются элементы и панели, необходимые для работы с программой. Многие инструменты имеют свойства, изменять которые можно с помощью панель настроек.

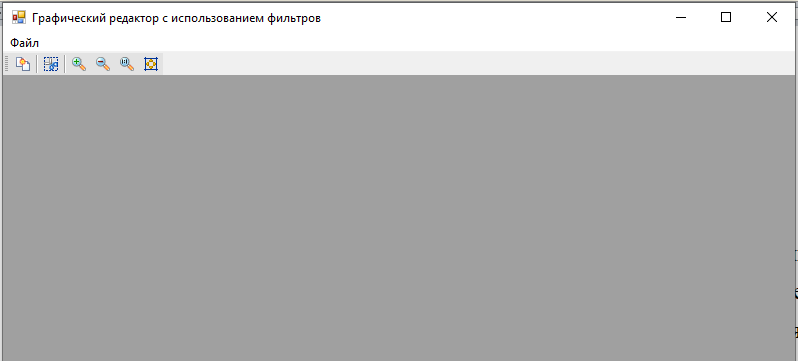


Рисунок 17 – Основная экранная форма

### 

### 2.2 Разработка и описание пользовательского меню

В современных программных системах количество функций, доступных пользователю достаточно велико, поэтому основным элементом управления является функциональное меню, которое строится по иерархическому принципу, т.е. взаимосвязанные функции группируются в отдельные пункты меню, которые доступны пользователю на верхнем уровне.

В разработанной системе меню состоит из трех пунктов: файл, изображение, фильтры. Структура меню представлена на рисунке 19.

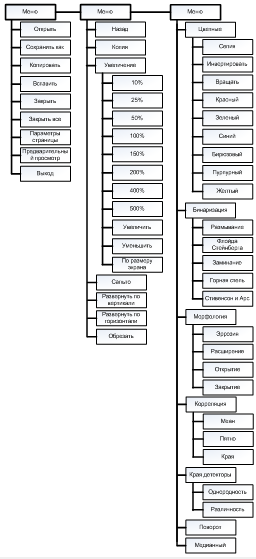


Рисунок 19 – Структура меню программы

При выборе пункта меню "Файл" открывается выпадающее меню, состоящее из нескольких подпунктов:

* "Открыть…" - открытие файла ранее созданного документа;
* "Сохранить как"- сохранение изображения в нужном формате;
* "Копировать"- копировать фрагмент изображения в буфер обмена;
* "Вставить"- вставить фрагмент изображения из буфер обмена;
* "Закрыть"- закрыть действующее изображение;
* "Закрыть все"- Закрыть все открытые изображения;
* "Выход" - выход из программы.

При выборе пункта меню "Изображение" открывается выпадающее меню, позволяющее пользователю изменить изображение или действие над ним. Выпадающее меню состоит из нескольких подпунктов:

* "Назад" - отменить действие;
* "Копия" - сделать копию изображения;
* "Увеличение" – при наведении появляется возможность выбора увеличения;
* "Сальто" – разворот изображения зеркально;
* "Разворот по вертикали" – разворот изображения по вертикали;
* "Разворот по горизонтали" – Разворот изображения по горизонтали.

При выборе пункта меню "Фильтры" открывается выпадающее окно, которое состоит из следующих подпунктов:

* "Цветные" - при наведении открывается выпадающее окно, которое предлагает на выбор цвет фильтра;
* "Бинаризация" - при наведении открывается выпадающее окно, которое предлагает на выбор фильтр бинаризации;
* "Морфология" - при наведении открывается выпадающее окно, которое предлагает на выбор фильтр морфологии;
* "Корреляция" - при наведении открывается выпадающее окно, которое предлагает на выбор фильтр корреляции;
* "Края детектор" - при наведении открывается выпадающее окно, которое предлагает на выбор фильтры;
* "Изменить размер" - изменить размер заданного изображения;
* "Поворот" - Повернуть изображение на заданный угол;
* "Медианный" – Медианный фильтр.

2.3 Описание тестового примера

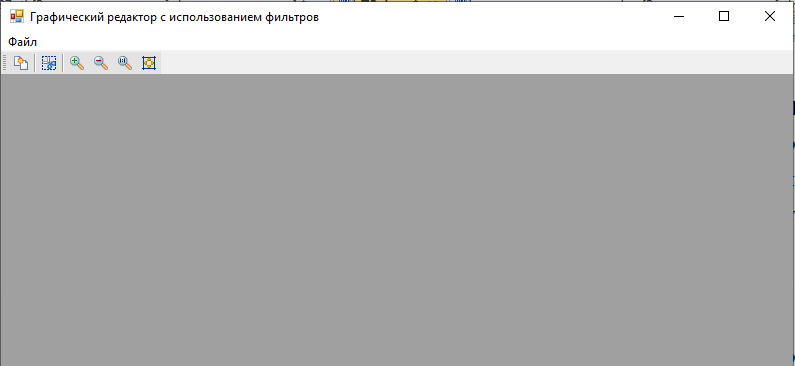
Рассмотрим работу программы. При запуске программы перед пользователем появляется окно, показанное на рисунке 20.

Рисунок 20 – Экранная форма создания нового изображения.

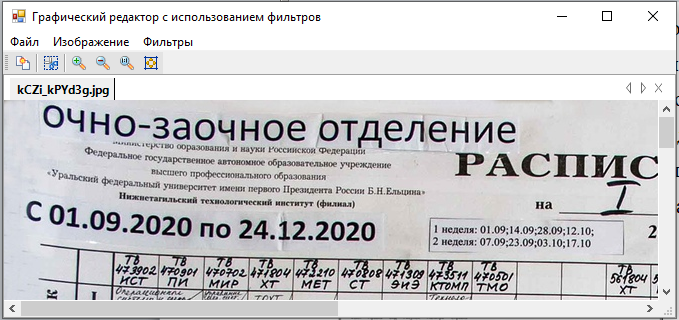
Для открытия ранее созданного изображения необходимо выбрать подпункт "Открыть" пункта меню "Файл", либо нажать на соответствующую кнопку на панели инструментов. При этом откроется диалоговое окно, позволяющее выбрать необходимый файл документа. Затем, после нажатия на кнопку "Открыть" в текстовом поле главной формы программы отобразится выбранной изображение. Экранная форма данного состояния программы отображена на рисунке 21.

Рисунок 21 – Экранная форма открытия документа

Для того чтобы сохранить изображение необходимо выбрать подпункт "Сохранить как" пункта меню "Файл. При этом откроется диалоговое окно, позволяющее сохранить изображение.

Возможность масштабирования изображения в пункте меню Увеличение, компонента Изображение на рис.22.

Предусмотрено режимы масштабирования в 10, 25, 50, 75, 100, 150, 200, 400, 500 процентов. Также можно изменять масштаб по шагам с помощью горячих клавиш Ctrl + 8 и Ctrl + 7. Предусмотрена возможность подгона масштаба изображения под размер экрана (Ctrl + 9).

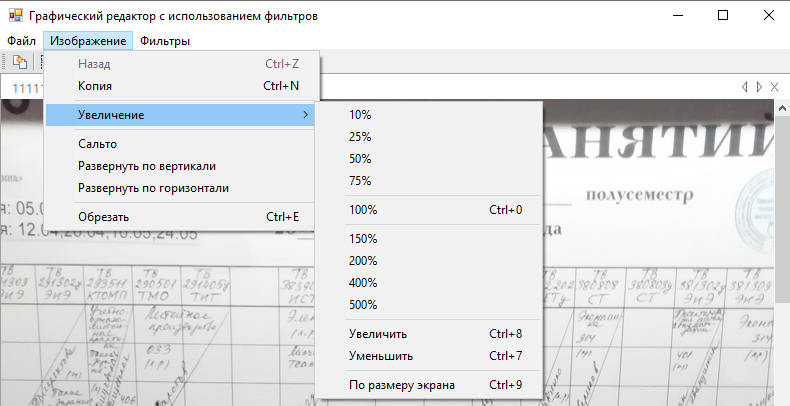


Рисунок 22 - Возможность масштабирования изображения.

Результат функции копирования изображения приведен на рис.23.

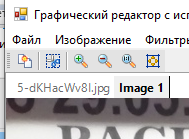


Рисунок 23 - Результат функции копирования изображения

Эта функция удобна, если пользователю после выполнения ряда действий по обработке графического изображения необходимо поэкспериментировать или протестировать какие-то эффекты, чтобы не навредить настройкой действующего образца. Создается соответствующая вкладка, которая содержит новую копию графического изображения.

Перечень реализованных в программном обеспечении опций из цветной фильтрации графического изображения в главном меню пользователя приведен на рис.24.

Поддерживаются возможности по преобразованию цветного изображения в оттенки серого, сепии (оригинал становится более тонированным, в коричневых тонах), инвертация цветов, извлечения красного, зеленого или синего каналов, вращения, и цветной фильтрации.

Дополнительно поддерживаются опции по редактированию графического изображения согласно фильтров цветов пурпурного, голубого и желтого, что придает соответствующие эффекты образцу.

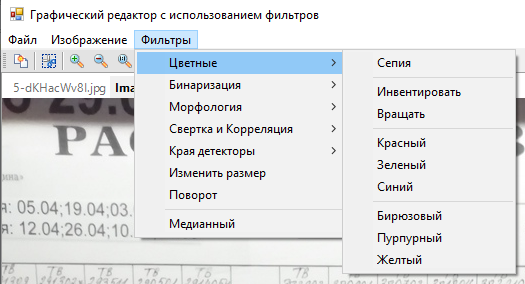


Рисунок 24- Перечень реализованных в программном обеспечении опций из цветной фильтрации графического изображения

Перечень реализованных в программном обеспечении опций по бинаризации графического изображения приведен на рис.25. В частности, поддерживается обычное размывания и размывание Байера, алгоритм размывания Флойда-Стейнберга и другие возможности.

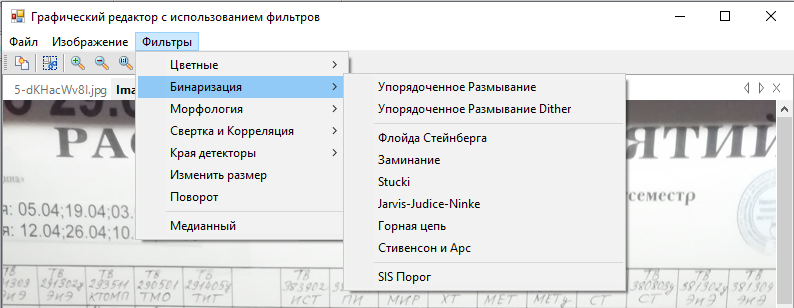


Рисунок 25 - Перечень реализованных в программном обеспечении опций с бинаризации графического изображения

Морфологическая обработка изображений представляет собой совокупность нелинейных операций, связанных с формой или морфологией признаков в изображении. Морфологические операции зависят только от относительного упорядочения значений пикселов, а не от их числовых значений, и, следовательно, подходят для обработки бинарных изображений.

Морфологические операции также могут быть применены к полутоновым изображениям таким образом, что их функции переноса света неизвестные и поэтому их абсолютные значения пикселей не имеют значительной важности. В частности, поддерживается возможность дилатации, которая использует выбран структурный элемент для расширения форм изображения.

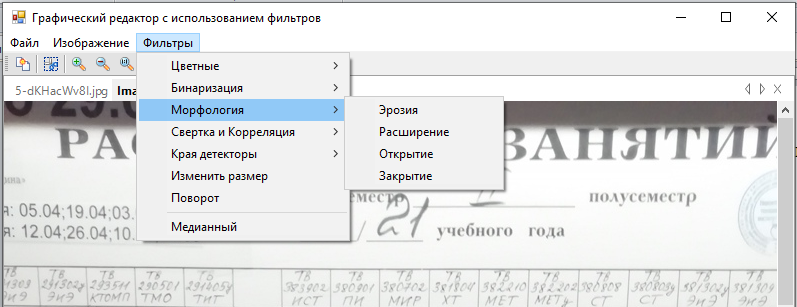


Рисунок 26 -Перечень реализованных в программном обеспечении опций с морфологической обработкой графического изображения

Перечень реализованных в программном обеспечении опций по выполнению свертки и корреляции графического изображения приведен на рис.27. Предусмотрена возможность наложения эффекта размытия, обострение.

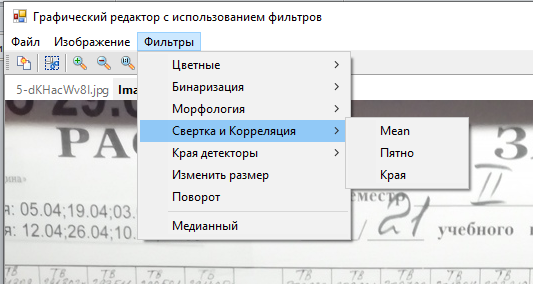


Рисунок 27- Перечень реализованных в программном обеспечении опций по выполнению свертки и корреляции графического изображения

Перечень реализованных в программном обеспечении опций по выявлению точек цифрового изображения приведен на рис.28.

Приведенные опции основанные на математических методах, направленных на выявление точек в цифровом изображении, при котором яркость изображения резко меняется или имеет разрывы. Точки, в которых изменения яркости изображения резко, как правило, организуются в набор криволинейных участков и называются краями.

В частности поддерживается метод Кенни для выявления края, использующий многоступенчатый алгоритм ​​для выявления широкого спектра кромок в изображениях. Также реализована поддержка метода Собела-Фельдмана для создания образа подчеркивания краев. С технической точки зрения это дискретный оператор дифференцирования, вычисления аппроксимации градиента функции интенсивности изображения. В каждой точке изображения, результат оператора Собела-Фельдмана есть или соответствующий вектор градиента или норма этого вектора. Метод основан на свертке изображения с небольшим диапазоном фильтра в горизонтальном и вертикальном направлениях, и, следовательно, относительно недорогой с точки зрения вычислений.

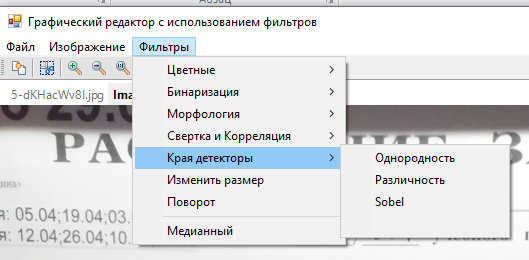


Рисунок 28 - Перечень реализованных в программном обеспечении опций по выявлению точек цифрового изображения

# Заключение

В ходе выполнения проекта был произведён анализ предметной области по методологии объектной декомпозиции, разработан UML-проект.

Было реализовано информационное программное обеспечение системы в соответствии с проектом. Подготовлены контрольные примеры, и проведено автономное и комплексное тестирование и отладка.

Систему отличает дружественный дизайн интерфейса, интуитивно понятное управление, быстрый отклик, возможность работы на различных версиях ОС Windows, подробное справка по программе и руководство пользователя.

# Список использованных источников

1. Учебник по информатике. Графический редактор [Электронный ресурс] – http://solidbase.karelia.ru/edu/zonna/3\_ychebnik\_8.htm (дата обращения 13.03.2014 г.).
2. Симонович С.В. Информатика. Базовый курс [Текст] / C.В. Симонович. 2-е изд. БХВ-Петербург, 2003. – 496 с.: ил.
3. Зеленко Л.С. Технологии программирования и программная инженерия (часть 1) [Текст]: учебное пособие / Л.С. Зеленко. – Самара: Изд-во Самарского государственного аэрокосмического университета, 2006. - 96 с.
4. Описание растрового изображение [Электронный ресурс] – http://ru.wikipedia.org/wiki/Расровая\_графика (дата обращения 27.03.2014 г.).
5. Диаграммы UML проекта [Электронный ресурс]. – http://ru.wikipedia.org/wiki/UML (дата обращения 28.04.2014 г.).
6. Иващенко А.В. Основы методологии проектирования автоматизированных систем обработки информации и управления [Текст]: учебное пособие / А.В. Иващенко. – Самара: СНЦ РАН, 2009. – 40 с., ил.
7. Проектирование пользовательского интерфейса [Электронный ресурс]. – http://www.williamspublishing.com/PDF/5-8459-0276-2/part7.pdf (дата обращения 10.05.2014 г.).
8. Языки программирования [Электронный ресурс]. – http://www.gpntb.ru/win/book/4/Doc4.HTML (дата обращения 28.04.2014 г.).
9. Хохлов А.Е. Основы информатики [Электронный ресурс]. – http://window.edu.ru/resource/967/53967/files/stup321.pdf (дата обращения 28.04.2014 г.).