**Приложения для интегрирования водяного знака в изображение**

В наше время множество людей выкладывают различные изображения в интернет. Они используются в разных сферах, начиная от ведения блога и заканчивая профессиональным заработком на творчестве. Однако, выкладывая личную работу в интернет, необходимо обезопасить себя и свои авторские права, поскольку без этого изображение могут украдут для другого сайта или продадут от своего имени. Специально для выхода из такой ситуации есть водяные знаки.

В связи с современными проблемами было решено создать программу для интегрирования водяного знака в изображение.

**1.1 Сравнительная характеристика существующих систем интегрирования водяного знака в изображение.**

Для такой цели уже существуют приложения аналоги, такие как:

**FastStone Photo Resizer**

Возможности программы: включение/отключение наложения водяного знака, указание файла, который будет использоваться в этих целях (то есть сам файл у вас должен быть подготовлен заранее), подгон его позиционирования относительно основы, регулировка прозрачности и пара эффектов. Доступна пакетная обработка, что очень актуально в условиях необходимости быстрого нанесения водяного знака на несколько файлов, например, скриншотов для инструкции.

Далее идёт:

**Easy Image Modifier**

Чтобы осуществить наложение водяного знака на одно или сразу несколько изображений в этом приложении, надо активировать это средство в разделе добавления водяного знака, напишите текст, который хотите видеть на изображении, укажите параметры шрифта и расположение знака. После этого останется нажать на кнопку «Обработать*»*. В этом приложении нельзя выбрать и наложить какое-то изображение из собственных ресурсов. В этом приложении доступны: поворот, отражение, изменение размера, конвертирование в некоторые распространенные форматы.

**BImage Studio** ещё один аналог

Первый вариант, текстовый, позволяет лишь только ввести слова, изменить цвет и стиль шрифта, указать позиционирование этой надписи. Второй вариант предлагает указать адрес файла с готовым водяным знаком на компьютере, изменить его размер и настроить место, куда он будет помещен.

Из дополнительных возможностей тут есть изменение размера, фильтры (яркость, контраст и пр.), обрезка, поворот и пара других мелких опций. Также здесь имеется пакетная обработка.

**1.2 Общая характеристика и особенности интегрирования водяного знака в изображение**

Из анализа приложений аналогов можно выделить самые важные возможности приложения для интегрирования водяного знака на изображение:

В приложении у пользователя должна быть возможность выбрать 2 картинки, которые он хочет преобразовать.

Далее у пользователя должна быть возможность изменить размер, прозрачность, и расположение водяного знака.

После должна быть возможность сохранить итоговое изображение в желаемом месте.

**1.3 Инструментальных средств разработки приложения для интегрирования водяного знака в изображение**

Выбор инструментальных средств лежит между **C++, java и С#**

C++ и Java являются языками программирования общего назначения, что означает, что можно создавать практически любые типы программных приложений практически для любых платформ.

C++ используется для создания операционных систем, настольных приложений, веб-браузеров, механизмов рендеринга веб-браузеров, библиотек машинного обучения, приложений с тяжелой графической обработкой, баз данных, встроенных систем, мобильных приложений и т.д.

Java используется для создания серверов приложений, веб-приложений, мобильных приложений, настольных приложений, модульных тестов, корпоративных приложений, игр, облачных приложений, веб-API и т.д.

Программы на C++ обычно работают быстрее, так как их не нужно интерпретировать, а программы на Java не так быстры, поскольку их нужно сначала интерпретировать, и это требует времени.

Хотя у С++ есть и недостатки перед Java. C++ не является безопасным для памяти языком. Вы можете управлять памятью, что является большим преимуществом. Но могут возникать ошибки памяти и серьезные проблемы, такие как сбои, во время выполнения программы.

**Сравнивая С++ и С#** можно сказать, что в С# огромным достоинством является возможность легкого создания пользовательского интерфейса, а также функция автоматической сборки мусора. Но несмотря на это большим минусом C# является то, что компиляция происходит по мере необходимости, то есть во время работы приложения могут быть подтормаживания.

Для наглядности представлена сравнительная таблица функционала языков

В итоге можно сделать вывод: наиболее подходящий язык программирования для реализации проекта является – C++.

**2 Цель и задачи курсового проекта**

Целью курсового проекта является разработка приложения для интегрирования водяного знака в изображение.

Для достижения цели составлены задачи, представленные на экране.

**3 Технологическая часть**

**3.1 Формализованное описание**

Для наложения знака, требуются 2 изображения (основное и водяной знак), далее требуется задать параметры водяного знака (размер, расположение и прозрачность) и после обработки полученных данных программа выдаст итоговое изображение.

**3.2 Постановка задачи обработки информации**

Задача обработки информации состоит в наложении водяного знака на изображение с возможностью изменять размер, расположение и прозрачность знака с помощью ползунков или счетчиков.

**3.3 Разработка функциональной структуры приложения для интегрирования водяного знака в изображение**

У пользователя будет доступ к 4 модулям графического интерфейса.

В первом пользователь сможет задать желаемые изображения, во втором пользователь сможет задать желаемые параметры водяного знака. Далее пользователь сможет просмотреть изображение и сохранить его.

Так же есть модули, скрытые от пользователя:

В первом таком модуле производится проверка корректности всех данных заданных пользователем, далее эти данные передаются в следующих модуль, который их обрабатывает, после чего они выводятся на экран пользователю.

**3.4 Разработка компонентов математического обеспечения приложения для интегрирования водяного знака в изображение**

Для наложения водяного знака на изображение создан класс ImageWork.

Для изменения размера водяного знака разработан метод changeSizeWatermark, в нём используются формулы, представленные на экране.

Для изменения прозрачности знака разработан метод changeTransparencyWatermark, который меняет прозрачность каждого пикселя в градации от 0 до 255.

Для изменения позиции знака разработан метод changePositionWatermark в котором используется метод DrawImage из класса Graphics.

**3.5 Разработка структуры интерфейса**

Структура пользовательского интерфейса состоит из 3 окон, по которым можно перемещаться, каждое окно предоставляет свой тип взаимодействия.

Пользователь сможет выбрать изображения, вбив путь к ним с помощью проводника или соответствующего текстового поля.

Далее пользователь сможет выбрать нужные параметры водяного знака при помощи ползунков или счетчиков и посмотреть полученное изображение в полноэкранном режиме двойным нажатием мыши на место отображения картинки.

В последнем окне реализоваеа возможность сохранить изображение с помощью проводника или текстового подя.

**3.8 Тестирование программного комплекса**

**В ходе тестирования было проверено:**

1)Приложение успешно загружает изображения

2)При некорректном пути выводится сообщение о ошибке

2)Водяной знак успешно накладывается на изображение

3) Итоговое изображение успешно сохраняется

4)При некорректном пути или названии файла выводится сообщение об ошибке

**ХАРАКТЕРИСТИКА ПРОГРАММНОГО И АППАРАТНОГО ОБЕСПЕЧЕНИЯ**

На слайде представлена структура программного обеспечения. Приложение разработано под управлением операционной системы Windows 10 на платформе Windows Forms в .NET 5. Средой разработки является Microsoft Visual Studio 2017.

В данной таблице приведена характеристика проблемно-ориентированного программного обеспечения.

А на этом слайде представлены требования к ЭВМ, необходимой для нормального функционирования программы.

**ВЫВОДЫ ПО КУРСОВОМУ ПРОЕКТУ**

Тоже что-то надо рассказать?