**Фоторедактор**

**Выполнил**: учащийся

9 информационного - математического класса

**Никешин Евгений**

**Пенза, 2019**

Цели и задачи.

**Цель:**

* Создать приложение фоторедактора;

**Задачи:**

* Проанализировать задачу;
* Определить возможности фоторедактора;
* Создать пользовательский интерфейс;
* Составить алгоритм работы редактора;
* Создать основные классы;
* Создать все функции редактора;
* Отладка программы;
* Написать пояснительную записку и подготовиться к защите проекта.

Актуальность

Сейчас существует большое множество различных редакторов, но они не всегда располагают всем необходимым интерфейсом, или сложны в обращении.

Этапы работы над проектом

* Постановка задачи;
* Создание простого интерфейса;
* Составление алгоритма проекта;
* Написание и отладка программы;
* Написание пояснительной записки и подготовка защите проекта.

Описание проекта

Ввод/Вывод

**Ввод:**

При активации программы фоторедактора, пользователю предоставляется окно через активные элементы которого он может воздействовать на фотографию.

**Вывод:**

При активации определённого элемента окна, фотография, загруженная в программу будет соответственно изменяться.

Функции и классы

При написании проекта были созданы следующие функции и классы:

* Photo\_Editor – основной класс программы, отвечает за взаимосвязь пользователя и изображения.
  + init() – конструктор класса Photo\_Editor, отвечает за взаимосвязь дизайна окна с функциями фоторедактора.
  + base() – коннектит активные элементы дизайна к соответствующим фунциям.
* Editor\_tools – класс, содержащий в себе все функции фоторедактора.
  + init() - конструктор класса Editor\_tools, отвечает за хранение основных данных текущего изображения и данных необходимых для работы функций.
  + copy\_picture() – отвечает за создание копии изображения;
  + new\_picture() – отвечает за подключение к фоторедактору новой картинки;
  + exit\_picture() – отвечает за вывод картинки в окно редактора;
  + roted\_l() – отвечает за поворот картинки против часовой;
  + roted\_r() – отвечает за поворот картинки по часовой;
  + sepia() – отвечает за фильтр изображения в формате сепия
  + negative() – отвечает за фильтр изображения в формате негатив
  + black() – отвечает за фильтр изображения в формате черно-белое
  + frame() – отвечает за создание вокруг изображения рамки
  + recoloring\_picture() – отвечает за изменение процентного содержания цвета в изображении
* Error – класс, отвечающий за предупреждение пользователя об ошибке.
  + init(message) – конструктор класса Error, отвечает за создание окна и вывод message.

Библиотеки

* PyQt5
* PIL
* os
* sys

Заключение

Мною был реализован фоторедактор, который может поворачивать изображение, применять к изображению фильтр, создавать копии изображений, изменять процентную составляющую цвета в изображении.