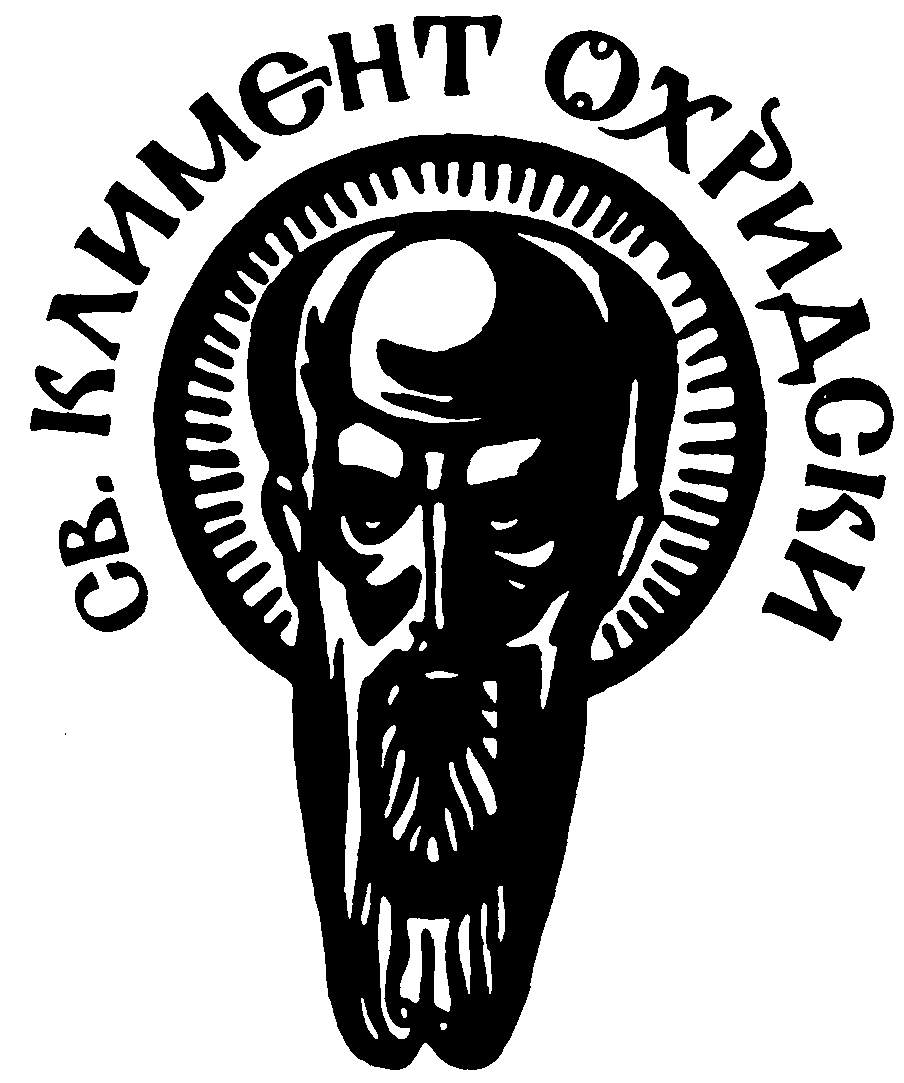
**Софийски университет „Св. Климент Охридски”**

**Факултет по математика и информатика**

****

**Обектно – ориентирано програмиране**

**Курсов проект на тема:**

**„Растерна графика“**

**Изготвила: Весела Илиянова Стоянова**

**Ф.Н. 71949**

**I курс Бакалавър**

**Специалност: Информационни системи**

**Ръководител: Калин Георгиев**

**07.06.2020 г.**

**Съдържание:**

**Таблици и графики:**

1. Увод
   1. Описание и идея на проекта

В рамките на този проект трябва да се разработи приложение, което представлява конзолен редактор на растерни изображения. Той трябва да поддържа различни файлове, стартиране на сесия, прилагане на различни трансформации върху изображенията и записване на резултатат.

* 1. Цел и задачи на разработката

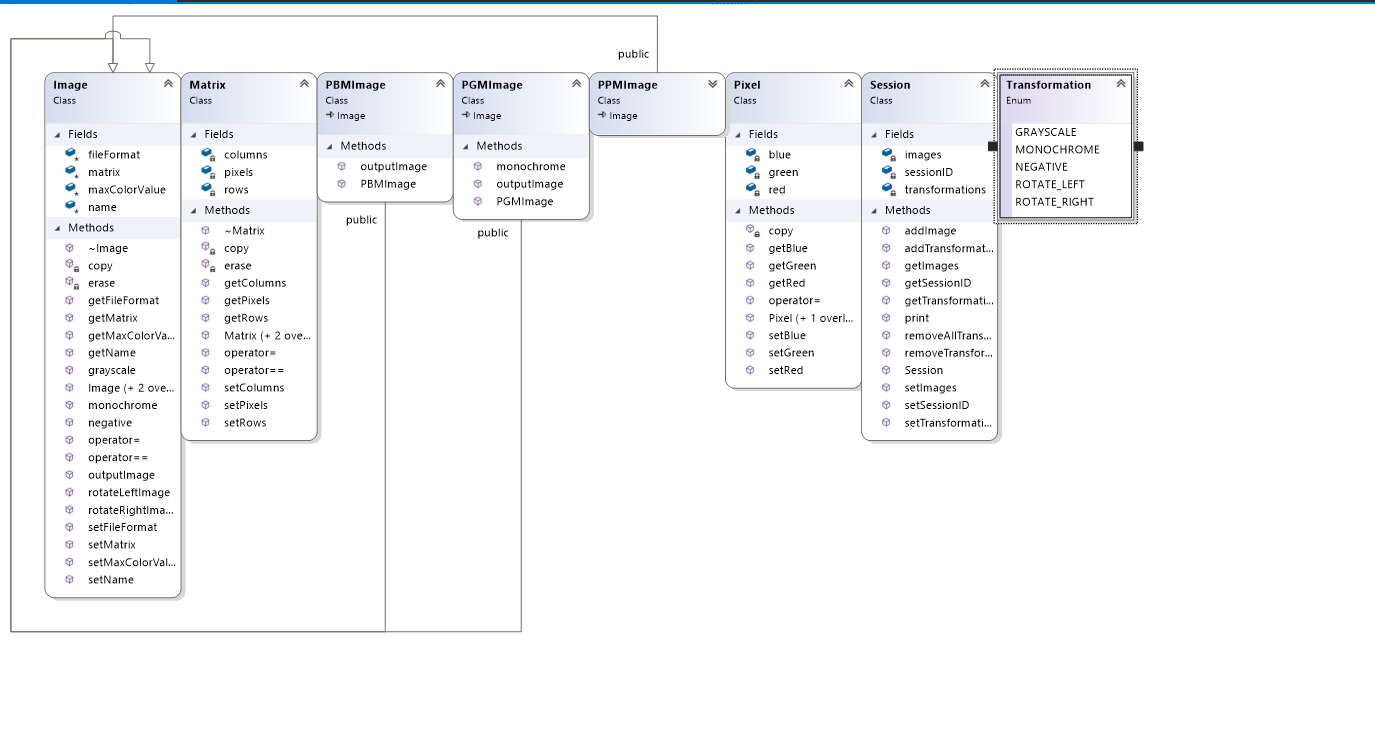
Основната цел е програмата да бъде достатъчно лесна и удобна за използване. За целта в проекта е реализиран CommandsExecutor, който съдържа метод showStartMenu. В main() функцията на програмата ще можем да стартираме програмата само с извикването на този метод. Програмата ще изисква от потребителя да въведе някоя от следните функции: load, help или exit, с помощта на която да определи коя функция да бъде изпълнена. При въвеждането на невалидни данни (данни, различни от load, help, exit) на стандартния изход ще бъдат изведени отново възможните функции. Всички промени, които се правят, трябва да бъдат записвани във файл. Когато бъде записан нов файл, изпълнението на програмата продължава и се извежда съобщение за успешното записване на данните във файл. Програмата завършва, когато бъде въведена функцията exit.

* 1. Структура на документацията

Структурата на документацията се състои в няколко опорни точки. Първоначално се представят належащите проблеми, както и търсеното на достатъчно ефективни решения. След това се преминава към проектиране на проекта и накрая се преминава към реализация и тестване на проекта. Накрая се представят насоки за бъдещо развитие и усъвършенстване.

1. Преглед на предметната област
   1. Основни дефиниции, концепции и алгоритми, които ще бъдат използвани

* Наследяване – начин за създаване на нови класове чрез използване на компоненти и поведения на съществуващи класове.
  1. Дефиниране на проблеми и сложност на поставената задача
  2. Подходи , методи (евентуално модели и стандарти) за решаване на поставените проблеми
  3. Потребителски(функционални) изисквания(права, роли, статуси, диаграми) и качествени(нефункционални) изисквания(скалируемост, поддръжка)

1. Проектиране
   1. Обща архитектура – ООП дизайн

.