

## **1. Минимальный комплект программной документации**

### **1.1. Техническое задание:**

В данной лабораторной работе разрабатывается программа "Менеджер договоров", предназначенная для сегментации и визуализации договоров на трудоустройство по компаниям и профессиям.

Программа должна загружать данные из файла, сегментировать их по компаниям и профессиям, а также предоставлять пользователю возможность визуализировать результаты в виде круговых диаграмм.

Программа должна быть написана на языке Python. Для построения интерфейса необходимо использовать библиотеку tkinter. Для построения графиков использовать библиотеку matplotlib.

Данные должны загружаться из файла, в котором каждая строка содержит компанию и профессию, разделённые запятой.

### **1.2. Пояснительная записка:**

Программа представляет собой простой графический интерфейс, который позволяет пользователю загружать данные о договорах на трудоустройство и анализировать их. Она поддерживает сегментацию данных по компаниям и профессиям с последующей визуализацией результатов в виде круговых диаграмм.

Логика работы заключается в следующем:

- 1) Пользователь загружает файл с договорами.
- 2) Программа проверяет формат данных.
- 3) Происходит сегментация договоров по компаниям или профессиям.
- 4) Результаты сегментации отображаются в виде круговой диаграммы.

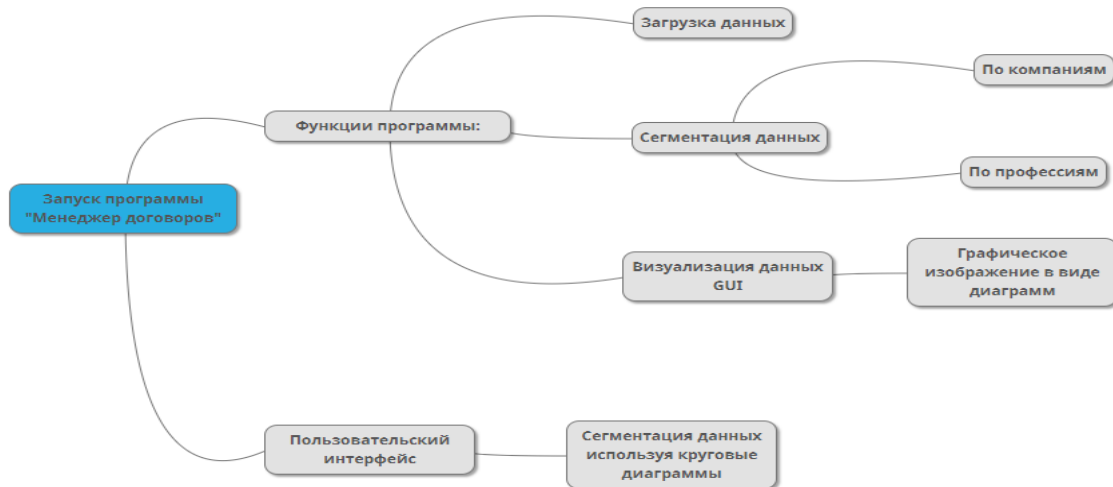
### **1.3. Руководство программиста:**

Программа состоит из следующих основных классов и методов:

- 1) Contract - класс, описывающий договоры
- 2) ContractManager - класс для работы с договорами
  - load\_contracts\_from\_file() - метод, который загружает данные из файла
  - segment\_by\_company() - метод, который сегментирует данные по компаниям
  - segment\_by\_profession() - метод, который сегментирует данные по профессиям
- 3) ContractApp - класс, описывающий GUI
  - load\_contracts() - метод, который загружает договоры
  - show\_company\_chart() - метод, который отображает диаграмму по компаниям
  - show\_profession\_chart() - метод, который отображает диаграмму по профессиям

## 2. Тестовая документация

### 2.1. Mind map:



### 2.2. Чек лист:

Проверка	Пример	Результат
Пустой файл	...	Ошибка, Файл пуст или содержит некорректные данные
Недопустимый формат файла	<u>ValveEngineer</u>	Ошибка, неверный формат данных
Правильный формат файла	<u>Valve, Engineer</u>	Успешно!

### 2.3. Набор тест-кейсов:

Тест кейс 1:

1.1: Файл содержит данные в формате «Компания,Профессия»

Шаги: Запустить программу

Нажать кнопку «Загрузить договор»

Выбрать файл с корректными данными

Ожидаемый результат: 1) Появляется сообщение об успешной загрузке: "Договоры успешно загружены"

2) Данные загружаются в программу и сегментируются

Тест кейс 2:

1.1: Файл содержит данные в неверном формате

Шаги: Запустить программу

Нажать на кнопку «Загрузить кнопку»

Выбрать файл с некорректными данными

Ожидаемый результат: 1) Выводится сообщение об ошибке: "Ошибка, неверный формат данных".

2) Данные не загружаются.