# Требования к проекту

## 1 Введение

Название проекта: Eisenhower Matrix

Матрица Эйзенхауэра — это один из способов краткосрочного планирования, который позволяет разбивать дела по категориям на основе приоритетов.

Границы проекта: проект будет представлять собой десктопное приложение, которое позволит пользователю вносить задачи в одно из четырех полей в зависимости от важности и срочности, устанавливать собственные приоритеты для задач, редактировать созданные задачи, удалять уже выполненные задачи. Также для каждой задачи можно будет указать время, до которого нужно выполнить задачу, и оповещения в тот момент, когда истекает срок выполнения задачи. Проект не будет содержать функционал, значительно выходящий за рамки создания, редактирования, удаления задач, а также уведомления при окончании срока выполнения задачи.

## 2 Требования пользователя

2.1 Программные интерфейсы

DB Browser for SQLite

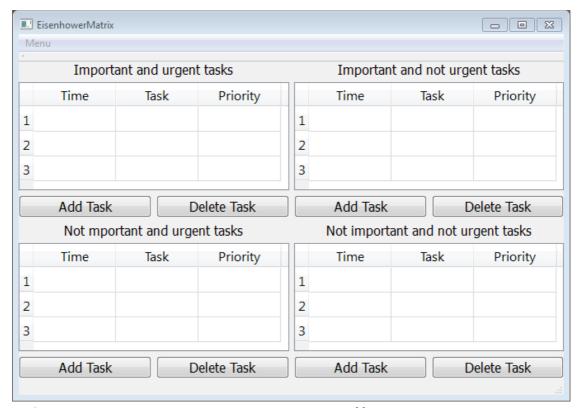
SQL Server

Version 3.4.0

SQLite Version 3.8.7

#### 2.2 Интерфейс пользователя

Приложение будет состоять из одного окна, в котором будут четыре таблицы, отображающие текущие задачи и их характеристики (время создания, срок выполнения поставленной задачи установленный пользователем приоритет). Каждая таблица снабжена кнопками, которые позволяют добавлять задачи в таблицу и удалять выделенные в таблице задачи.



#### 2.3 Характеристики пользователей

Программа рассчитана на широкий круг пользователей, которые могут иметь различный уровень образования, опыт и техническую грамотность.

#### 2.4 Предположения и зависимости

На изменение требований, изложенных в данном документе могут повлиять следующие факторы:

- Изменение круга пользователей приложения или появление большого числа пользователей, желающих иметь дополнительный функционал в приложении.
- Переход на другие платформы (web или mobile).

## 3 Системные требования

## 3.1 Функциональные требования

Приложение должно:

- 1. Иметь GUI, с помощью которого можно вводить, редактировать и удалять задачи.
- 2. Сохранять задачи, которые ввел пользователь, в базе данных и восстанавливать эту информацию при очередном открытии приложения.
- 3. Иметь систему оповещений в том случае, если истекает срок выполнения задачи.

## 3.2 Нефункциональные требования

- 3.2.1 Кроссплатформенность. Приложение должно работать на разных операционных системах. Чтобы решить эту задачу, для написания приложения будет использоваться язык программирования Java. Для создания GUI будет использоваться JavaFX.
- 3.2.2 Надежность. Приложение должно выводить правильные данные. Должно быть предусмотрено большинство возможных исключительных ситуаций. В случае исключительных ситуаций данные не должны быть утеряны. Время обработки подавляющего большинство запросов не должно превышать 1 секунды.