**Проект «калькулятор»**

**Список требований к проекту калькулятора**

▎1. **Функциональные требования**

• Основные арифметические операции: Калькулятор должен поддерживать сложение, вычитание, умножение и деление.

• Дополнительные функции: Возможность выполнять операции с процентами, квадратные корни и возведение в степень.

• История операций: Пользователь должен иметь возможность просматривать историю предыдущих вычислений.

• Поддержка скобок: Возможность выполнения операций с использованием круглых скобок для изменения порядка вычислений.

• Копирование и вставка: Пользователь должен иметь возможность копировать результаты и вставлять их в другие приложения.

• Поддержка различных форматов ввода: Калькулятор должен принимать как десятичные, так и дробные числа.

▎2. **Нефункциональные требования**

• Интерфейс пользователя: Должен быть интуитивно понятным, простым и удобным для использования.

• Кроссбраузерная совместимость: Калькулятор должен работать во всех современных веб-браузерах (Chrome, Firefox, Safari и т.д.).

• Мобильная адаптивность: Интерфейс должен быть адаптирован для работы на мобильных устройствах.

• Производительность: Время отклика калькулятора должно быть минимальным, даже при сложных вычислениях.

• Безопасность: Защита данных пользователя, включая предотвращение возможных уязвимостей.

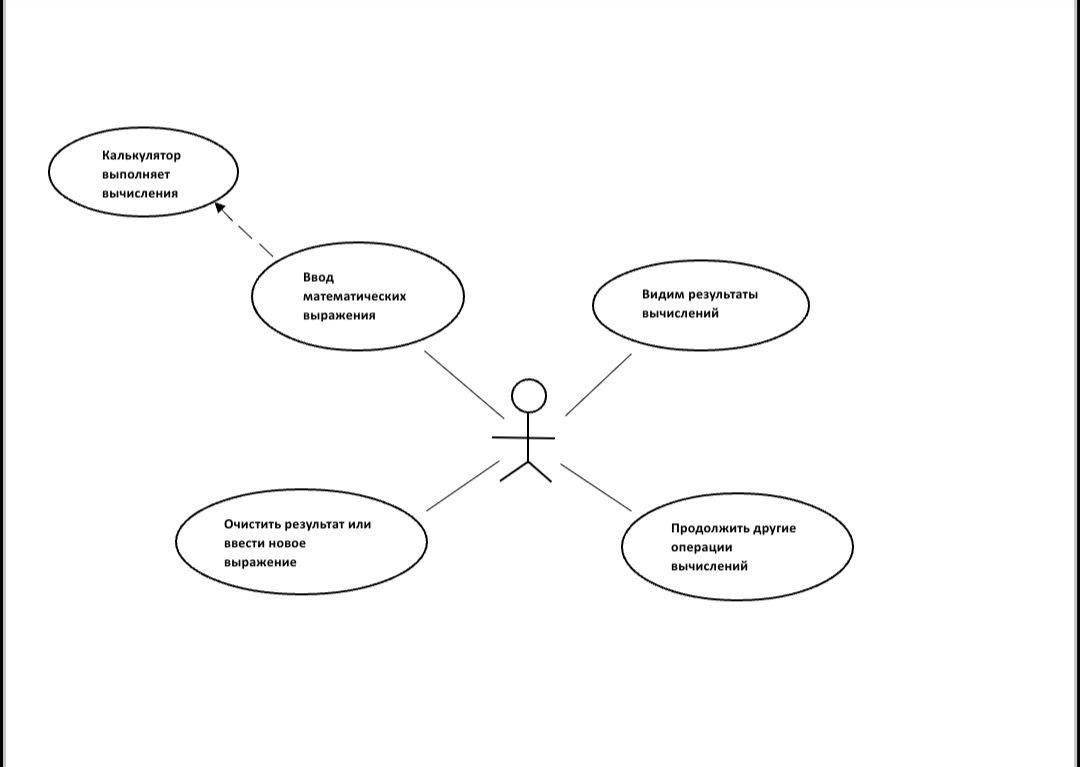
▎3. **Требования к тестированию**

• Юнит-тесты: Все функции калькулятора должны быть протестированы с использованием юнит-тестов.

• Тестирование интерфейса: Проверка на наличие ошибок в пользовательском интерфейсе и взаимодействии с ним.

• Тестирование производительности: Оценка скорости работы калькулятора под нагрузкой.

**Диаграмма вариантов использования**



**Итоги проекта по созданию калькулятора**

1. Цели и задачи:

• Определение, были ли достигнуты первоначальные цели проекта.

• Оценка выполнения поставленных задач, таких как разработка интерфейса, реализация функций и тестирование.

2. Технические аспекты:

• Описание технологий и инструментов, использованных в проекте (язык программирования, библиотеки, фреймворки).

• Оценка производительности калькулятора: скорость выполнения операций, стабильность работы.

3. Отзыв пользователей:

• Сбор отзывов от пользователей: что им понравилось, а что можно улучшить.

• Анализ отзывов для выявления возможных направлений для доработки.

4. Временные рамки:

• Сравнение фактических временных затрат с запланированными.

**Итоги:**

Эти шаги позволяют создать функциональный и надежный калькулятор, который сможет выполнять основные математические операции с точностью и эффективностью.