**1. Краткая спецификация разрабатываемого модуля**

**Входные данные:**

* Номер заявки (цифровой)
* Дата добавления заявки (дата/время)
* Вид авто (строка)
* Модель авто (строка)
* Описание проблемы (строка)
* ФИО клиента (строка)
* Номер телефона клиента (строка)
* Статус заявки (перечисление: новая заявка, в процессе ремонта, завершена)
* Этап выполнения (перечисление: готова к выдаче, в процессе ремонта, ожидание автозапчастей)
* Ответственный за выполнение работ (строка)
* Комментарии автомеханика (строка)
* Заказанные автозапчасти и материалы (строка)

**Выходные данные:**

* Список заявок с параметрами:
  + Номер заявки
  + Статус заявки
  + Дата добавления
  + ФИО клиента
  + Модель авто
  + Описание проблемы
* Уведомления о смене статуса заявки
* Статистика по выполненным заявкам:
  + Количество выполненных заявок
  + Среднее время выполнения заявки
  + Статистика по типам неисправностей

**2. Основной алгоритм решения учета заявок на ремонт автомобилей**

В основном алгоритме будут использоваться следующие процессы:

1. **Добавление заявки**: При добавлении заявки в базу данных, происходит валидация входных данных и создание новой записи в базе.
2. **Редактирование заявки**: Возможность изменить статус, этап выполнения или описание заявки.
3. **Отслеживание статуса**: По запросу пользователя отображается список заявок с их текущими статусами.
4. **Назначение ответственного**: Автомеханики назначаются на заявку, и вносят данные о выполненных работах.
5. **Статистика**: По завершении заявок система рассчитывает количество выполненных заявок, среднее время выполнения и типы неисправностей.

**3. Детализация одной функции (расчет количества заявок и среднего времени ремонта)**

**Алгоритм расчета количества заявок:**

1. Подключение к базе данных.
2. Выполнение запроса на подсчет заявок, которые имеют статус "завершена".
3. Возвращение результата — количество заявок.

**Алгоритм расчета среднего времени ремонта:**

1. Подключение к базе данных.
2. Выполнение запроса на получение времени начала и окончания каждой заявки с статусом "завершена".
3. Для каждой заявки вычисляется время выполнения как разница между датой завершения и датой начала.
4. Подсчитывается среднее значение по всем заявкам.

**4. Блок-схема основного алгоритма**

Для создания блок-схемы в соответствии с ГОСТ 19.701, нужно представить основное взаимодействие между пользователем и системой:

1. **Начало**.
2. **Добавление заявки**:
   * Ввод данных (номер, дата, вид авто, описание и т. д.).
   * Валидация данных.
   * Сохранение заявки в базу данных.
3. **Редактирование заявки**:
   * Выбор заявки по номеру.
   * Изменение статуса или других данных.
4. **Отслеживание статуса заявки**:
   * Запрос списка заявок.
   * Отображение статуса.
5. **Назначение ответственного**:
   * Назначение автомеханика.
   * Внесение комментариев.
6. **Расчет статистики**:
   * Подсчет количества заявок.
   * Расчет среднего времени выполнения.
7. **Конец**.

**5. Текстовое описание алгоритма (ГОСТ 24.301)**

**Алгоритм подсчета количества заявок:**

1. Подключение к базе данных.
2. Выполнение запроса для выбора всех заявок, имеющих статус "завершена".
3. Подсчет количества таких заявок.
4. Вывод результата.

**Алгоритм расчета среднего времени ремонта:**

1. Подключение к базе данных.
2. Выполнение запроса для получения времени начала и окончания заявок со статусом "завершена".
3. Для каждой заявки вычислить разницу между временем завершения и начала.
4. Подсчитать общее время выполнения всех заявок.
5. Разделить на количество заявок, чтобы получить среднее время.