УТВЕРЖДЕН

А.В.00001-01 33 01-1-ЛУ

**Программный модуль для учета заявок на ремонт автомобилей**

**Руководство системного программиста**

А.В.00001-01 33 01-1-ЛУ

**Листов 30**

2024

**АННОТАЦИЯ**

В данном программном документе приведено руководство системного программиста по настройке и использованию программного модуля, предназначенного для автоматизации и учёта заявок на ремонт автомобилей, назначения исполнителей и контроля выполнения работ.

Оформление программного документа «Руководство системного программиста» произведено по требованиям ЕСПД (ГОСТ 19.101-77, ГОСТ 19.103-77, ГОСТ 19.104-78, ГОСТ 19.105-78, ГОСТ 19.106-78, ГОСТ 19.503-79, ГОСТ 19.604-78).

Содержание

[1. Общие сведения о программе 4](#_Toc185374731)

[2. Структура программы 5](#_Toc185374732)

[3. Настройка программы 7](#_Toc185374733)

[4. Проверка программы 8](#_Toc185374734)

[5. Дополнительные возможности 24](#_Toc185374735)

[6. Сообщение системному программисту 25](#_Toc185374736)

[7. Перечень принятых сокращений 27](#_Toc185374737)

# **Общие сведения о программе**

Программное обеспечение предназначено для повышения эффективности управления заявками на ремонт автомобилей в автосервисе, управления заявками, отслеживания текущего статуса ремонта.

Данное программное обеспечение может быть использовано для станций технического обслуживания, занимающихся ремонтом автомобилей, для повышения эффективности работы автомехаников и операторов, а также улучшения взаимодействия с заказчиками.

В данном документе ниже будут описаны следующие сведения о программном обеспечении:

1. структура программы;
2. настройка и запуск программы;
3. проверка программы;
4. сообщения системному программисту.

# **Структура программы**

Программное обеспечение включает в себя различные модули (рабочие окна), которые разделены в зависимости от ролей пользователей.

Первым модулем является авторизация. Этот компонент отвечает за безопасный доступ пользователей к системе. Он осуществляет проверку учетных данных — логина и пароля — через базу данных. Если первая попытка входа оказывается неудачной, активируется механизм капчи для защиты от автоматизированных атак. После трех неудачных попыток система временно блокирует пользователя на три минуты. Как только вход проходит успешно, капча отключается. Все попытки входа, как удачные, так и неудачные, фиксируются в журнале входов.

Модуль авторизации предоставляет возможность отслеживать все попытки входа в систему, включая логин, дату, время и статус. Он оснащён функциями фильтрации по логину и сортировки по времени попыток, что позволяет пользователю видеть, была ли каждая попытка успешной или нет.

Модуль клиента дает заказчикам возможность создавать новые заявки на ремонт автомобиля, а также просматривать и редактировать уже существующие. Каждая заявка содержит информацию о типе и модели автомобиля, описании проблемы, а также ФИО и телефон клиента, и статус заявки. Фильтрация и сортировка заявок упрощает работу с большим массивом данных.

Модуль оператора предоставляет операторам доступ ко всем заявкам, с возможностью их фильтрации по статусу, типу автомобилей и другим критериям. Операторы могут назначать автомехаников для выполнения задач и изменять статус заявок, а также просматривать полную информацию по каждой из них.

Модуль мастера даёт автомеханикам возможность управлять заявками, которые им назначены. Механик может видеть описание проблемы, тип и модель автомобиля, а также данные клиента. После завершения ремонта, мастер обновляет статус заявки с указанием даты завершения и имеет доступ к информации о среднем времени выполнения задач, что помогает оценить его работу.

# **Настройка программы**

Для работы программного обеспечения требуется наличие компьютера с минимальными техническими характеристиками, а также стандартных периферийных устройств, таких как клавиатура и мышь. Установка клиентской части программы осуществляется через инсталлятор, после чего ПО готово к использованию.

Таблица 1 – Требования ПО.

|  |  |
| --- | --- |
| Процессор | Intel или AMD |
| ОЗУ | Не менее 1 ГБ |
| Накопитель | Минимум 30 МБ свободного пространства |
| Поддержка ОС | Windows или macOS |
| Видеокарта | Стандартная интегрированная видеокарта, поддержка графических интерфейсов |

Клиентская часть программы устанавливается через инсталлятор, после чего можно пользоваться ПО.

# **Проверка программы**

После установки программы на компьютер и ее запуска на экране появляется окно авторизации пользователя с предложением ввести логин и пароль. Внешний вид экранной формы авторизации показан на рисунке 1.

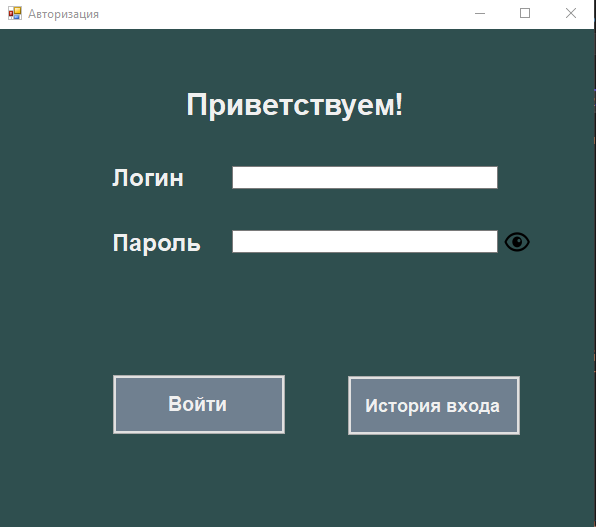


Рисунок 1 – Окно авторизации

Окно авторизации содержит следующие элементы:

1. Текстовое поле для ввода логина – предназначено для ввода уникального идентификатора пользователя.
2. Текстовое поле для ввода пароля – ввод пароля осуществляется в скрытом режиме (символы заменяются точками для защиты информации).
3. Кнопка «Войти» – проверяет введенные данные на корректность. Если логин и пароль введены правильно, система предоставляет доступ к интерфейсу пользователя в зависимости от его роли. В случае некорректных данных выводится сообщение об ошибке.
4. Кнопка «История входа» – открывает экранное окно с таблицей, в которой отображена история входов.
5. Капча (если активна) – в случае нескольких неудачных попыток авторизации появляется капча для предотвращения автоматизированных попыток входа. Пользователь должен ввести сгенерированный системой код из капчи для подтверждения, что он не является ботом.
6. Счетчик попыток – после нескольких неудачных попыток (например, двух) программа блокирует учетную запись на определенное время (например, 3 минуты). Система выводит сообщение о блокировке и информирует, когда можно будет повторить попытку входа.
7. Кнопка «Показать/скрыть пароль» – позволяет пользователю временно отобразить или скрыть введенный пароль, чтобы убедиться в его правильности.

Проверка модуля авторизации включает:

1. Вход с корректными данными – при вводе верного логина и пароля система должна перенаправить пользователя на соответствующее рабочее окно (например, окно заказчика, оператора или мастера).
2. Вход с некорректными данными – при вводе неверного логина или пароля система должна вывести сообщение об ошибке и увеличить счетчик неудачных попыток. При достижении предела попыток должна включаться защита (например, капча или блокировка на время).
3. Работа капчи – после первой неудачной попытки входа появляется капча, которую пользователь должен ввести для подтверждения, что он не бот. Система должна проверять корректность ввода капчи и при необходимости генерировать новый код.
4. Блокировка учетной записи – после нескольких неудачных попыток (например, двух) система блокирует возможность входа на заданный период времени. Проверяется правильность отображения сообщения о блокировке и возобновление работы после истечения времени блокировки.

После успешной/неуспешной авторизации данные об этой попытке сохраняются в базу данных, и пользователь может просмотреть их через окно истории авторизаций, предварительно нажав на соответствующую кнопку на форме авторизации. На экране истории отображаются данные о логине пользователя, времени попытки и ее статусе (успешно или ошибка). Внешний вид окна представлен на рисунке 2.

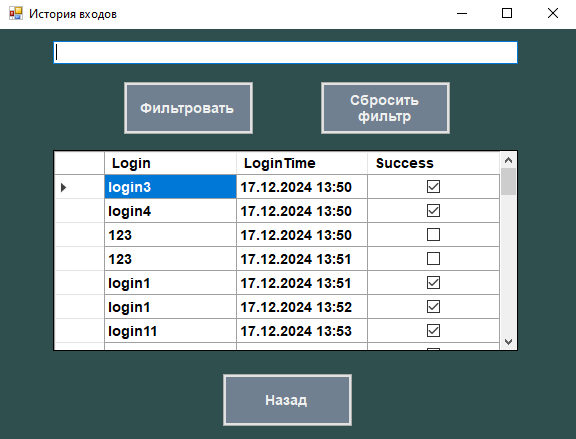


Рисунок 2 – окно истории входов

На экране истории входов представлены следующие элементы: поле фильтрации по логину, где можно ввести часть или полный логин для быстрого поиска конкретного пользователя, и таблица с логами авторизаций, содержащая столбцы:

1. Логин – отображает логин пользователя, который пытался войти.
2. Дата и время – показывает дату и время каждой попытки входа.
3. Статус – указывает, была ли попытка успешной или неуспешной.

Проверка этого модуля заключается в том, чтобы убедиться, что данные обо всех попытках входа корректно записываются в таблицу и отображаются в окне истории, а фильтрация по логину работает корректно.

После авторизации с помощью данных заказчика открывается окно пользователя. Экранный вид формы показан на рисунке 3.

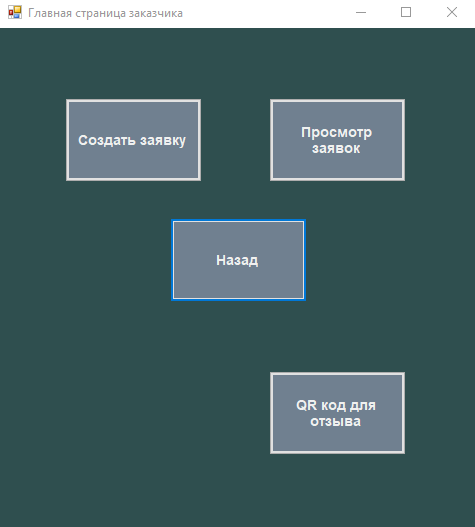


Рисунок 3 – Окно заказчика

Окно клиента содержит следующие элементы:

1. Кнопка «Создать заявку» – при нажатии открывается форма для добавления новой заявки клиента. В форму передается идентификатор клиента для корректной работы с заявками. Текущая форма остаётся открытой.
2. Кнопка «Просмотр заявок» – при нажатии открывается форма для просмотра существующих заявок клиента. Идентификатор клиента передается для отображения заявок именно этого клиента. Текущая форма закрывается.
3. Кнопка «Назад» – предназначена для выхода из учетной записи клиента и открытия формы авторизации. При нажатии закрывается текущая форма и открывается форма авторизации.
4. Кнопка «QR код для отзывы» – при нажатии появляется QR-код для перехода в Google Forms.

Проверка модуля формы клиента включает:

1. Добавление заявки – при нажатии кнопки «Добавить заявку» система должна корректно открыть форму для создания новой заявки и передать в неё идентификатор клиента. Проверяется корректность передачи данных и отображение формы без ошибок.
2. Просмотр заявок – при нажатии кнопки «Просмотр заявок» система должна закрыть текущую форму и открыть форму для просмотра заявок клиента. Проверяется корректность передачи идентификатора клиента и отображение списка заявок этого клиента.
3. Выход из учетной записи – при нажатии кнопки «Выйти» система должна закрыть текущую форму и открыть форму авторизации. Проверяется правильность завершения работы текущей формы и открытия формы авторизации без задержек и ошибок.
4. Проверка корректности передачи данных – проверяется, что при каждом действии (добавление, просмотр заявок) идентификатор клиента передаётся в новые формы без изменений, что обеспечивает работу с правильными данными.
5. Закрытие текущей формы – проверяется, что при открытии форм для просмотра заявок и выхода из учетной записи, текущая форма закрывается без зависаний или других проблем.

После нажатия на главной форме заказчика кнопки «Создать заявку» открывается форма для создания заказчиком заявки. Экранное окно формы представлено на рисунке 4.

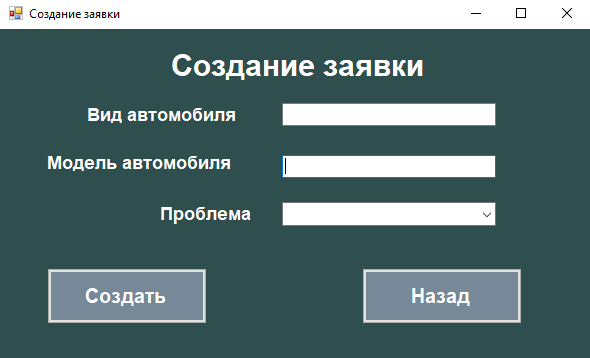


Рисунок 4 – Окно создания заявки

Форма добавления заявки содержит следующие элементы:

1. Текстовое поле для ввода вида автомобиля – предназначено для ввода типа автомобиля, который требует ремонта. Пользователь должен ввести значение вручную.
2. Текстовое поле для ввода модели автомобиля – используется для ввода модели автомобиля. Вводится вручную.
3. Выпадающий список для выбора проблемы – предоставляет пользователю список распространённых проблем с автомобилем. Пользователь выбирает одну из предложенных опций, таких как «Отказали тормоза», «Проблемы с электроникой» и др.
4. Кнопка «Создать» – сохраняет новую заявку. При нажатии кнопки происходит проверка корректности введённых данных и сохранение заявки в базу данных.
5. Кнопка «Назад» – закрывает форму добавления заявки без сохранения введённых данных.

Проверка модуля формы добавления заявки включает:

1. Проверка валидации данных – при нажатии кнопки "Добавить" система должна убедиться, что все поля заполнены корректно. Проверяется, что текстовые поля не пустые, выбран пункт в выпадающем списке.
2. Проверка списка проблем – проверяется, что выпадающий список содержит все предустановленные проблемы и пользователь может выбрать любую из них.
3. Проверка добавления автомобиля – система должна проверить, существует ли уже автомобиль с введёнными типом и моделью в базе данных. Если нет, автомобиль должен быть добавлен.
4. Проверка добавления заявки – при успешной валидации данных заявка должна быть добавлена в таблицу заявок. Проверяется корректность записи даты создания заявки, описания проблемы, статуса, идентификатора клиента и автомобиля.
5. Сообщение о результате – при успешном добавлении заявки система должна уведомить пользователя сообщением об успешной передаче заявки оператору. В случае ошибки отображается соответствующее сообщение с текстом ошибки.
6. Корректное закрытие формы – после успешного добавления заявки форма должна быть закрыта. Проверяется, что форма закрывается корректно без зависаний или ошибок.
7. Проверка корректного соединения с базой данных – проверяется, что соединение с базой данных устанавливается успешно, а операции вставки выполняются корректно.

После нажатия на главной форме заказчика кнопки «Просмотр заявок» открывается форма просмотра заявок заказчика. Экранное окно формы представлено на рисунке 5.

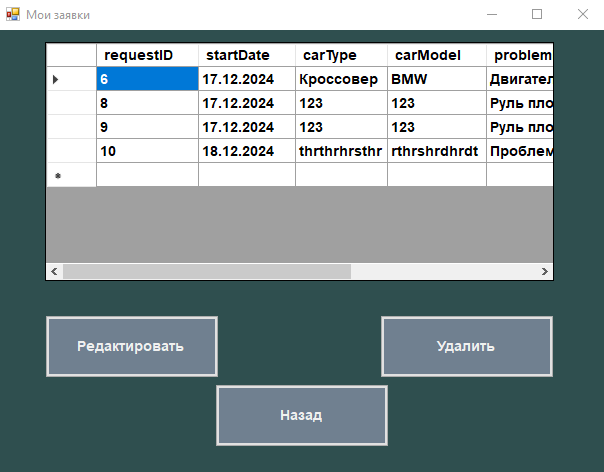


Рисунок 5 – Окно просмотра заявок заказчика

Форма просмотра заявок клиента содержит следующие элементы:

1. Таблица для отображения списка заявок клиента. Каждая строка содержит информацию о заявке, такую как дата создания, тип и модель автомобиля, описание проблемы, и статус заявки.
2. Кнопка «Редактировать» — открывает форму для редактирования выбранной заявки. Если заявка не выбрана, выводится предупреждение.
3. Кнопка «Удалить» — удаляет выбранную заявку после подтверждения. Если заявка не выбрана, выводится предупреждение.
4. Кнопка «Назад» — закрывает текущую форму и возвращает пользователя к форме клиента.

Основная цель проверки данного модуля состоит в том, чтобы подтвердить, что информация по всем заявкам корректно вносится в таблицу и отображается правильно, а также что при нажатии на кнопку редактирования открывается окно редактирования и можно выбрать нужную запись. Кроме того, необходимо обеспечить возможность удаления выбранной заявки с предварительным уведомлением пользователя о процессе удаления.

При нажатии на кнопку «Редактировать» в окне просмотра заявок заказчиком, открывается экранная форма редактирования заявки, представленная на рисунке 6.

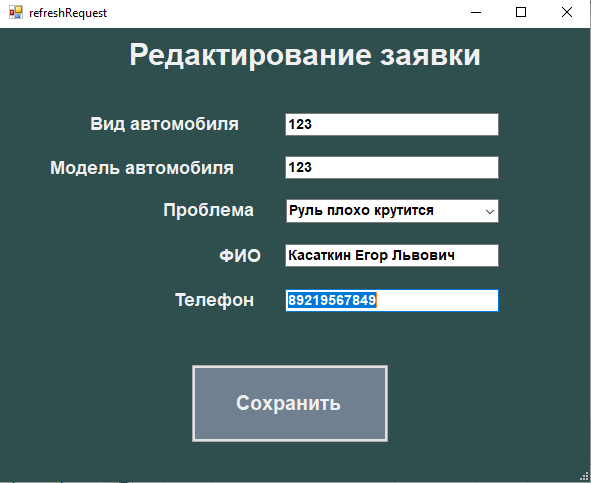


Рисунок 6 – Окно редактирования заявки

Элементами формы являются те же поля, что и на форме добавления заявки, и кнопка «Сохранить» которая обновляет выбранную заявку и закрывает форму редактирования.

Проверка этого модуля заключается в том, чтобы удостовериться в корректном обновлении записей.

После авторизации с помощью данных оператора открывается окно оператора. Экранный вид формы показан на рисунке 7.

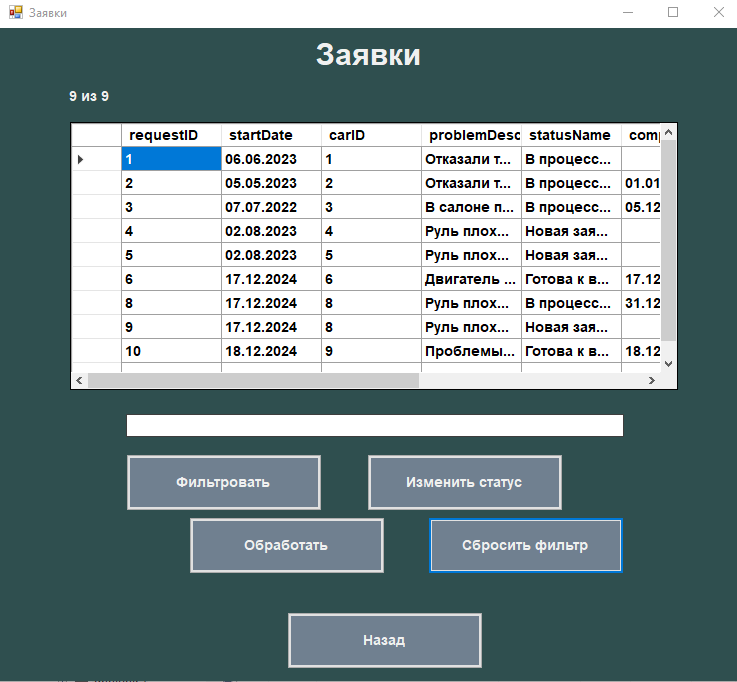


Рисунок 7 – Окно управления заявками оператора

Форма оператора содержит следующие элементы:

1. Таблица для отображения списка заявок. Каждая строка содержит информацию о заявке, такую как дата создания, тип и модель автомобиля, описание проблемы, и статус заявки.
2. Кнопка «Обработать» — открывает форму для обработки выбранной заявки. Если заявка не выбрана, выводится предупреждение.
3. Кнопка «Изменить статус» — открывает форму для изменения статуса выбранной заявки. Если заявка не выбрана, выводится предупреждение.
4. Текстовое поле для ввода фильтров.
5. Кнопки «Фильтровать» и «Сбросить фильтр» – первая отображает записи по заданному фильтру, вторая – сбрасывает фильтр и показывает все заявки.
6. Кнопка «Назад» — закрывает текущую форму и возвращает пользователя к форме клиента.

Проверка этого модуля заключается в том, чтобы убедиться, что данные при обработке и изменении статуса заявок отображаются корректно, а также корректной фильтрации и сортировке данных.

Экранная форма для обработки заявки оператором, которая открывается при выборе заявки и нажатии на соответствующую кнопку, представлена на рисунке 8.

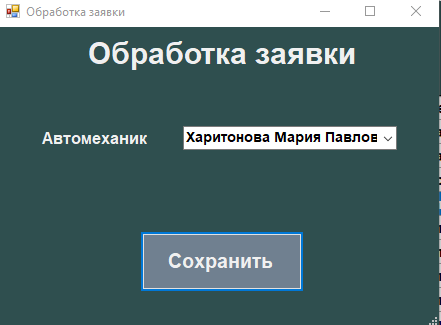


Рисунок 8 – Окно обработки заявок

Форма обработки заявки содержит следующие элементы:

1) Выпадающий список для назначения механика.

2) Кнопка «Сохранить» — сохраняет в таблице измененные данные о назначенном автомеханике и статусе заявки, закрывает форму обработки.

Проверка этого модуля заключается в том, чтобы убедиться, что после нажатия кнопки для заявки назначен автомеханик и статус заявки изменен на «В процессе».

Экранная форма для изменения статуса заявки оператором, которая открывается при выборе заявки и нажатии на соответствующую кнопку, представлена на рисунке 9.

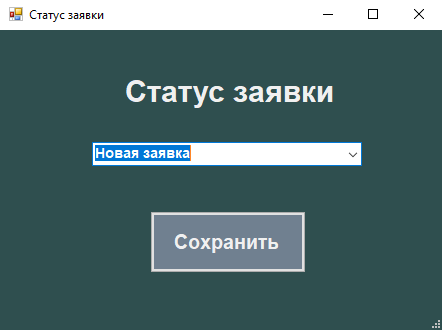


Рисунок 9 – Окно изменения статуса

Форма обработки заявки содержит следующие элементы:

1. Выпадающий список для выбора статуса.
2. Кнопка «Сохранить» — сохраняет в таблице измененные данные о статусе заявки.

Проверка этого модуля заключается в том, чтобы убедиться, что после нажатия кнопки статус заявки изменён на выбранный.

После авторизации с помощью данных автомеханика открывается окно автомеханика. Экранный вид формы показан на рисунке 10.

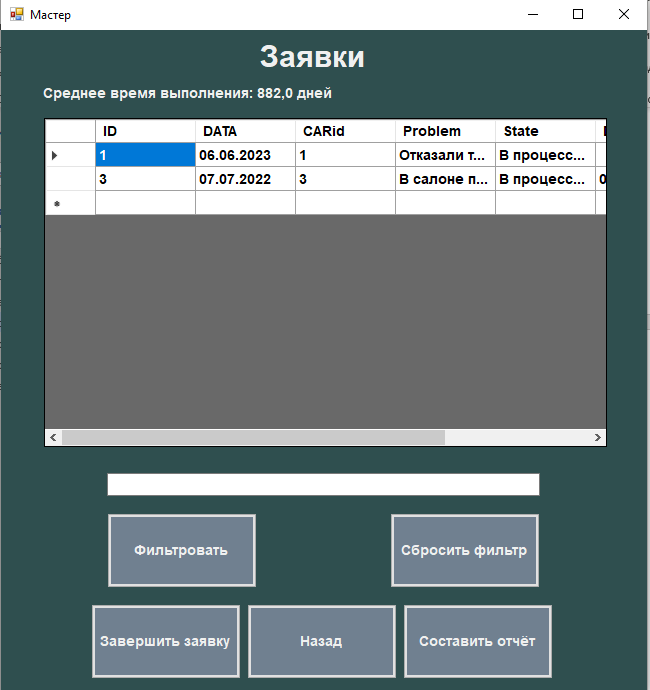


Рисунок 10 – Окно заявок автомеханика

Форма автомеханика содержит следующие элементы:

1. Таблица для отображения списка заявок автомеханика. Каждая строка содержит информацию о заявке, такую как дата создания, тип и модель автомобиля, описание проблемы, и статус заявки.
2. Кнопка «Завершить заявку» — изменяет статус выбранной заявки на «Готов к выдаче» и устанавливает дату выполнения заявки. Если заявка не выбрана, выводится предупреждение.
3. Кнопка «Составить отчет» — создает отчет по выбранной заявке в формате .txt. Если заявка не выбрана, выводится предупреждение.
4. Текстовое поле для ввода фильтров.
5. Кнопки «Фильтровать» и «Сбросить фильтр» – первая отображает записи по заданному фильтру, вторая – сбрасывает фильтр и показывает все заявки.
6. Кнопка «Назад» — закрывает текущую форму и возвращает пользователя к форме клиента.

Проверка этого модуля заключается в том, чтобы убедиться, что данные при завершении заявки отображаются корректно, составляется отчет по выбранной заявке, а также корректной фильтрации и сортировке данных.

После авторизации с помощью данных автомеханика открывается окно автомеханика. Экранный вид формы показан на рисунке 11.

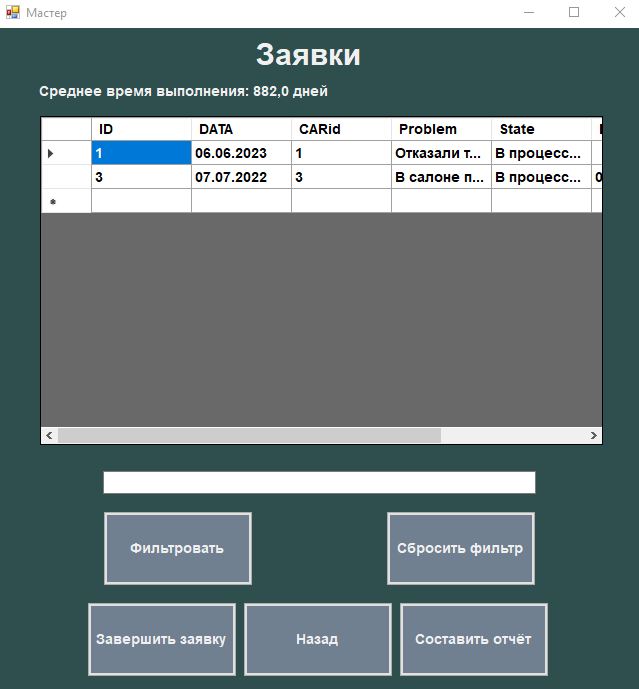


Рисунок 34 – Окно заявок автомеханика

Форма автомеханика содержит следующие элементы:

1. Таблица для отображения списка заявок автомеханика. Каждая строка содержит информацию о заявке, такую как дата создания, тип и модель автомобиля, описание проблемы, и статус заявки.
2. Кнопка «Завершить заявку» — изменяет статус выбранной заявки на «Готов к выдаче» и устанавливает дату выполнения заявки. Если заявка не выбрана, выводится предупреждение.
3. Кнопка «Составить отчет» — создает отчет по выбранной заявке в формате .txt. Если заявка не выбрана, выводится предупреждение.
4. Текстовое поле для ввода фильтров.
5. Кнопки «Фильтровать» и «Сбросить фильтр» – первая отображает записи по заданному фильтру, вторая – сбрасывает фильтр и показывает все заявки.
6. Кнопка «Назад» — закрывает текущую форму и возвращает пользователя к форме клиента.

Проверка этого модуля заключается в том, чтобы убедиться, что данные при завершении заявки отображаются корректно, составляется отчет по выбранной заявке, а также корректной фильтрации и сортировке данных.

После авторизации с помощью данных менеджера открывается окно менеджера. Экранный вид формы показан на рисунке 12.

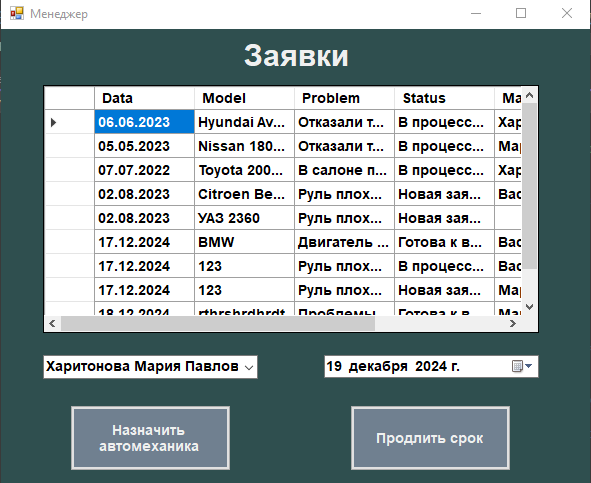


Рисунок 12 – Окно менеджера

Форма Менеджера содержит следующие элементы:

1. Таблица для отображения списка всех заявок. Каждая строка содержит информацию о заявке, такую как дата создания, тип и модель автомобиля, описание проблемы, и статус заявки.
2. Кнопка «Назначить автомеханика» — изменяет назначенного автомеханика выбранной заявки на выбранного из выпадающего списка. Если заявка не выбрана, выводится предупреждение.
3. Кнопка «Продлить срок» — изменяет срок завершения заявки на выбранную дату в календаре, изменяет статус заявки на «В процессе». Если заявка не выбрана, выводится предупреждение.
4. Выпадающий список для выбора назначаемого автомеханика.
5. DateTimePicker для продления даты завершения заявки.

Проверка данного модуля включает в себя подтверждение корректности отображения данных при назначении автомеханика и продлении срока выполнения запроса.

# **Дополнительные возможности**

В программном обеспечении не предоставляется никаких дополнительных возможностей для работы.

# **Сообщение системному программисту**

Программное обеспечение поддерживает отображение ошибок во время выполнения. Наиболее частые внештатные ситуации, возникающие при работе программы, отслеживаются и отображаются в виде всплывающих окон, уведомляющих об ошибке. Ниже представлено несколько рисунков с примерами окон, сообщающих об ошибке.

На рисунке 13 представлено окно с ошибкой, сообщающей о введении неверного логина и/или пароля.

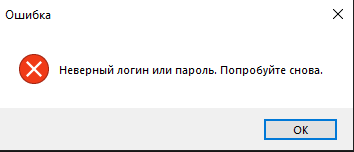


Рисунок 13 – Ошибка при введении неверных данных для входа в систему

На рисунке 14 представлено окно с ошибкой, сообщающей о введении неверной капчи.

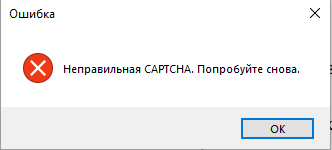


Рисунок 14 – Ошибка при введении неверной капчи

На рисунке 15 представлено окно с ошибкой, сообщающей о введении некорректных данных (пустые или заполненные некорректно поля) при заполнении/редактировании заявки.

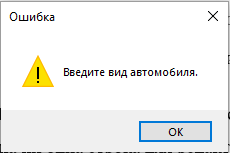


Рисунок 15 – ошибка при создании заявки

На рисунке 16 представлено окно с ошибкой, сообщающей о том, что не выбрана ни одна строка для редактирования.

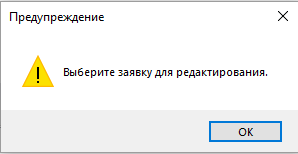


Рисунок 16 – Ошибка при попытке отредактировать заявку без выбранной строки

# **Перечень принятых сокращений**

**ГОСТ Р** – Государственный стандарт Российской Федерации

**ИСО** – Международная организация по стандартизации

**ПО** – Программное обеспечение

**СУБД** – Система управления базами данных