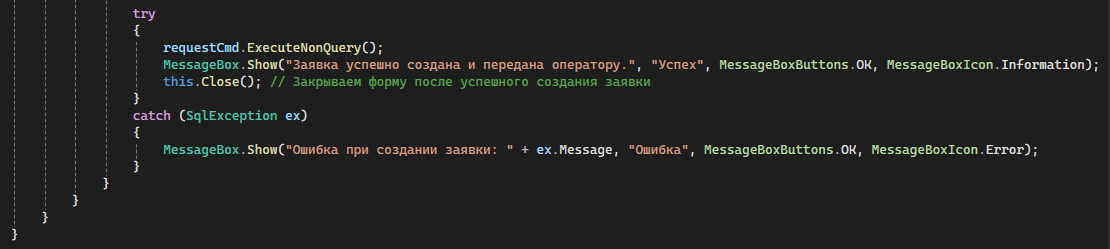
## **Качественные характеристики кода**

### Признаки хорошего кода – это совокупность характеристик и черт, которые демонстрируют, насколько эффективно написан программный код с точки зрения его продуктивности, ясности, доступности к изменению и других свойств. Код, соответствующий высоким стандартам, облегчает его обслуживание, оптимизирует работу и минимизирует число багов в дальнейшем использовании.

### **Полнота обработки ошибочных данных**

Код программы обеспечивает обработку ошибок при работе с базой данных. Он отвечает за создание заявки в БД и обработку любых ошибок, которые могут возникнуть в этом процессе.

### **Наличие тестов для проверки допустимых значений входных данных**

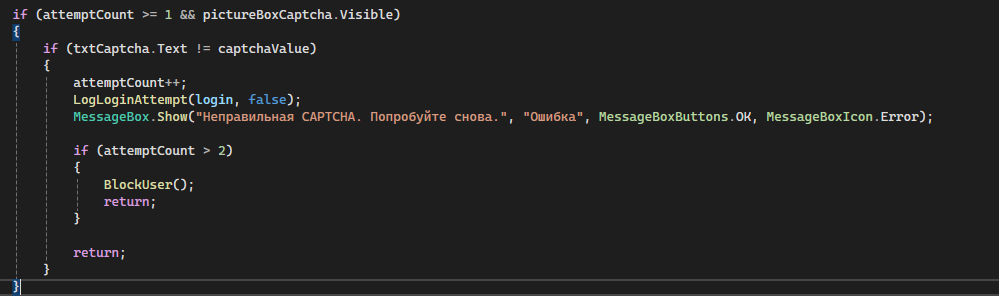
В коде реализован процесс проверки корректности данных, поступающих из формы авторизации. К примеру, введенные логин и пароль проходят верификацию на предмет их существования в базе данных через

вызов метода ValidateUser.



### **Наличие средств контроля корректности входных данных**

В коде присутствует множество проверок корректности входных данных, например, неверно введенной капчи.



### **Наличие средств восстановления при сбоях оборудования**

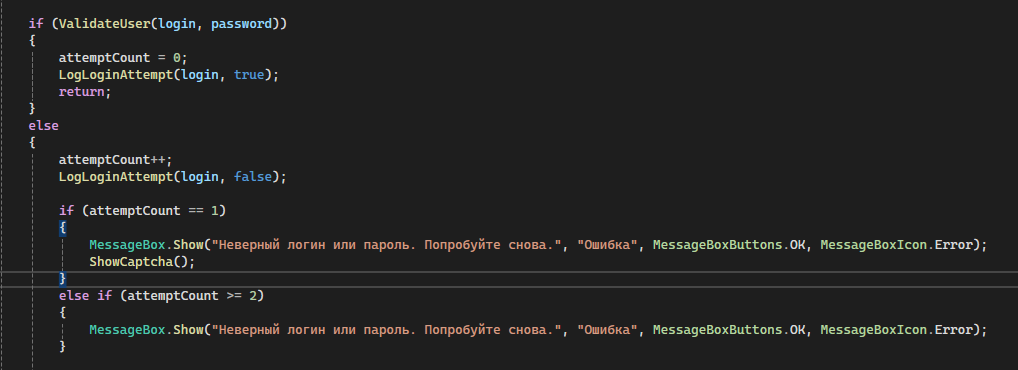
Код не содержит прямых методов восстановления после сбоев оборудования. Но, обработка исключений позволяет частично реагировать на проблемы с оборудованием, например, на сбои при подключении к базе данных. В случае сбоя система выводит сообщение об ошибке и указывает пользователю, что делать.

### **Наличие комментариев**

В коде присутствуют комментарии для объяснения методов и функций.

### **Наличие проверки корректности передаваемых данных**

Программный код обеспечивает надежную проверку достоверности данных, вводимых пользователем, включая логин, пароль и капчу. Контроль осуществляется как на этапе ввода информации, так и путем последующей валидации через запросы к базе данных.



### **Наличие описания основных функций**

Каждый метод в коде выполняет свою конкретную функцию, что согласуется с принципами организации и удобочитаемости программного кода. Ниже представлены основные методы программы с их подробным описанием.

Метод button1\_Click (отправка данных) является обработчиком события нажатия кнопки “Отправить”. Он осуществляет проверку корректности введенных данных, добавляет сведения о транспортном средстве и запросе в базу данных, а также отображает сообщения об успешной операции или возникших ошибках.

Метод button1\_Click(авторизация) обрабатывает нажатие кнопки входа в систему. Он проверяет, не заблокирован ли пользователь, проводит валидацию введенных данных, управляет процессом авторизации и отображает CAPTCHA при необходимости.

Метод button2\_Click(удаление заявки) является обработчиком события нажатия кнопки удаления заявки. Он проверяет, выбрана ли хотя бы одна заявка в таблице DataGridView, запрашивает подтверждение удаления у пользователя, вызывает метод удаления записи из БД, обновляет данные в таблице, и, если ни одна запись не выбрана, показывает предупреждающее сообщение.

Метод UpdateRequest(обновление заявки) используется для обновления данных заявки. Он выполняет проверку корректности введенных данных с помощью функции ValidateForm(), устанавливает соединение с базой данных и последовательно выполняет три обновляющих запроса: изменяет информацию об автомобиле, данные клиента и описание проблемы в заявке. После успешного выполнения всех запросов выводит сообщение об успехе, обновляет данные в родительской форме и закрывает текущую форму.