Министерство образования и молодежной политики Свердловской области

ГАПОУ СО «Екатеринбургский колледж транспортного строительства»

Отчёт по учебной практике

УП 01.01

Выполнила: Горелова Э.А

Группа: ПР-31

Преподаватель: Мирошниченко Г.В.

2024

1 Задание №1 Индивидуальное задание

[1.1 Описание задачи 3](#_Toc118960340)

[1.2 Описание разработанных функций 3](#_Toc118960342)

[1.3 Алгоритм решения 3](#_Toc118960343)

[1.4 Используемые библиотеки 3](#_Toc118960344)

[1.5 Тестовые случаи 3](#_Toc118960345)

[1.6 Используемые инструменты 3](#_Toc118960346)

[1.7 Описание пользовательского интерфейса 3](#_Toc118960347)

[1.8 Приложение (pr screen экранов) 3](#_Toc118960348)

# 1. **Задание №3 Индивидуальное задание**

1.1 Описание задачи:

Используя методы Linq и коллекции (использовать не менее 2-х видов

коллекций)

Описать базовый класс с указанными в задании полями и методами:

* Конструктор;
* Добавить 2 своих поля;
* Функция, которая определяет качество объекта – Q по заданной

формуле;

* Вывод информации об объекте.
* Реализуйте и проверьте методы для добавления и удаления

объекта базового класса (сделать 3 перегрузки!);

Построить класс потомок, который содержит:

* Дополнительное поле Р;
* Добавить еще 3 свойства
* Функция, которая определяет «качество» объекта класса потомка
* Добавить свою функцию для класса потомка

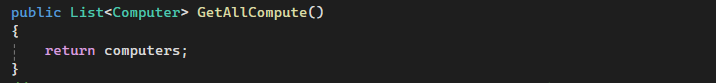
Создать проект для демонстрации работы: ввод-вывод информации об

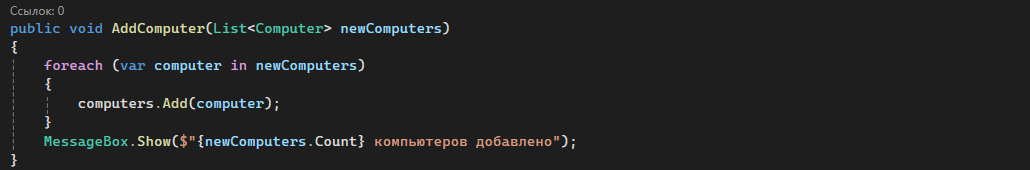
объектах базового класса и класса потомка.

Разработать модульные тесты к реализованным методам классов.

1.2 Описание разработанных функций

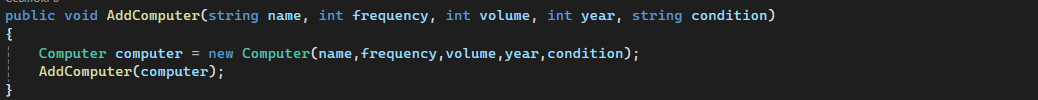
Класс Computer:

Функция GetAllCompute() предназначена для получения списка всех объектов типа Computer, которые хранятся в коллекции computers.Метод возвращает список объектов типа Computer (List<Computer>), который представляет собой коллекцию всех компьютеров, хранящихся в переменной computers. Если в коллекции нет компьютеров, метод вернёт пустой список.

Метод AddComputer(List<Computer>newComputers) для добавления новых объектов типа Computer в существующую коллекцию computers. Метод принимает один параметр: List<Computer> newComputers. Это список объектов типа Computer, которые необходимо добавить в коллекцию.



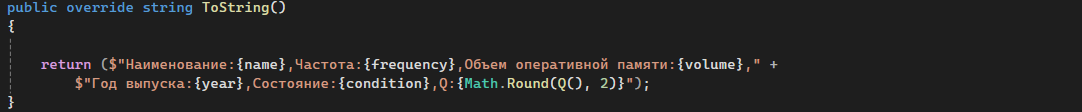
Метод AddComputer(Computer comp) для добавления одного объекта типа Computer в существующую коллекцию computers. После добавления метод отображает сообщение через MessageBox.Show(), информируя пользователя о том, что компьютер был добавлен.



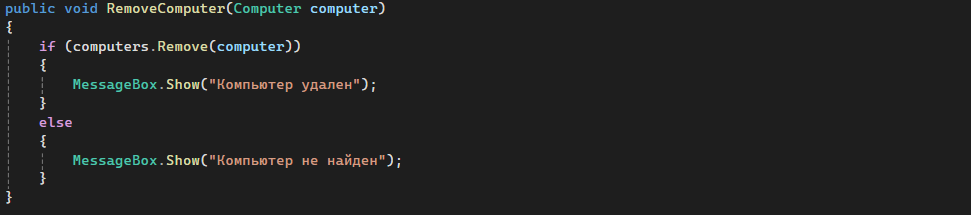
Метод AddComputer(string name, int frequency, int volume, int year, string condition) для создания и добавления объекта в список. Включает отображение сообщения, то пользователь увидит уведомление о добавлении компьютера.



Функция Q(), которая определяет качество объекта – Q по заданной формуле. Входными данными являются частота и объем оперативной памяти, выходными данными являются значение Q по заданной формуле.

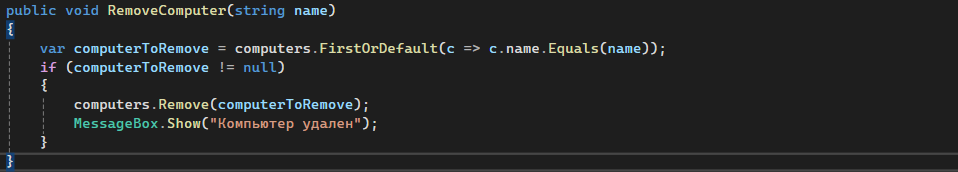


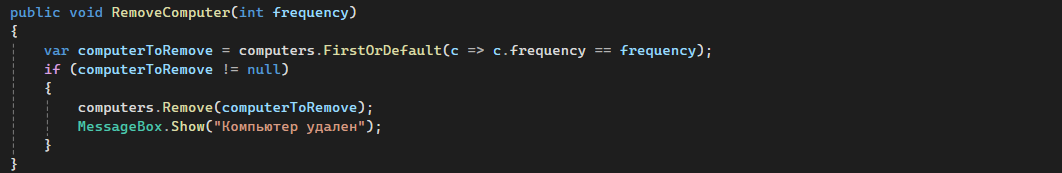
Функция ToString() для корректного вывода строки.



Метод RemoveComputer(Computer computer) предназначен для удаления объекта типа Computer из коллекции computers. Он проверяет, существует ли указанный компьютер в списке, и если да, то удаляет его, уведомляя пользователя об этом. В зависимости от результата операции, метод выводит одно из следующих сообщений:

* Если компьютер успешно удален: "Компьютер удален".
* Если компьютер не найден в списке: "Компьютер не найден".

Метод RemoveComputer(string name) предназначен для удаления объекта типа Computer из коллекции computers по его имени. Метод принимает один параметр string name: имя компьютера, который необходимо удалить из списка. Если компьютер с указанным именем найден и успешно удален, выводится сообщение: "Компьютер удален".

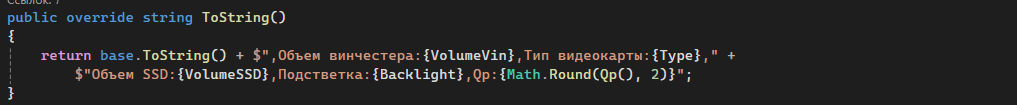


Метод RemoveComputer(int frequency) предназначен для удаления объекта типа Computer из коллекции computers по частоте. Метод принимает один параметр int frequency: частота компьютера, который необходимо удалить из списка. Если компьютер с указанным именем найден и успешно удален, выводится сообщение: "Компьютер удален".

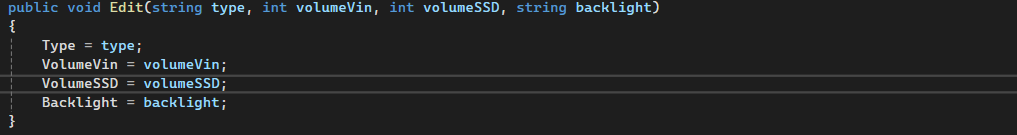
Класс GamingComputer:



Функция Qp(), вычисляет новое значение, объединяя результат другой функции, входные данные переменная VolumeVin, которая также должна быть double, выходными данными являются значение Qр по заданной формуле.



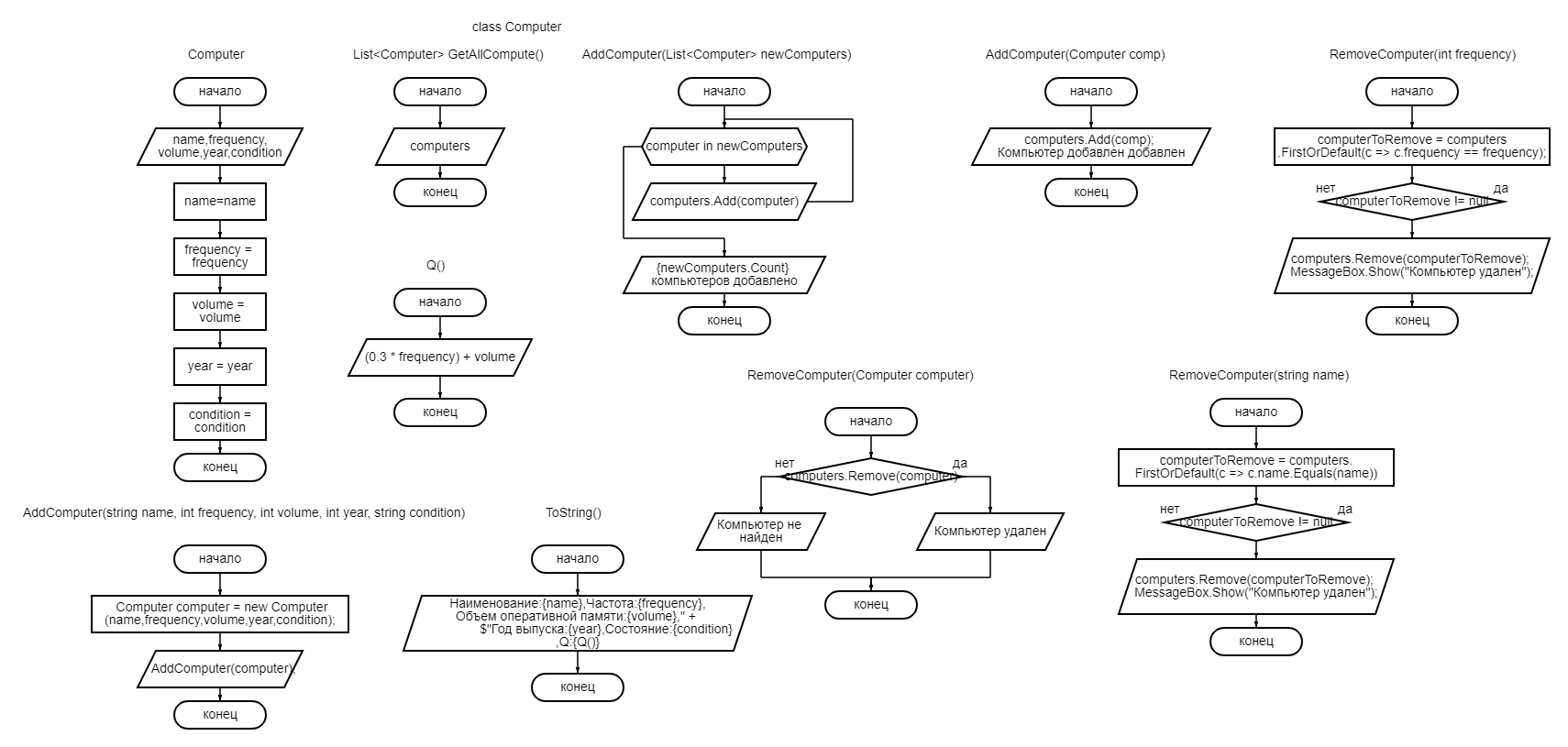
Функция ToString() для корректного вывода строки. Вызова base.ToString(), который извлекает строковое представление по умолчанию из базового класса.



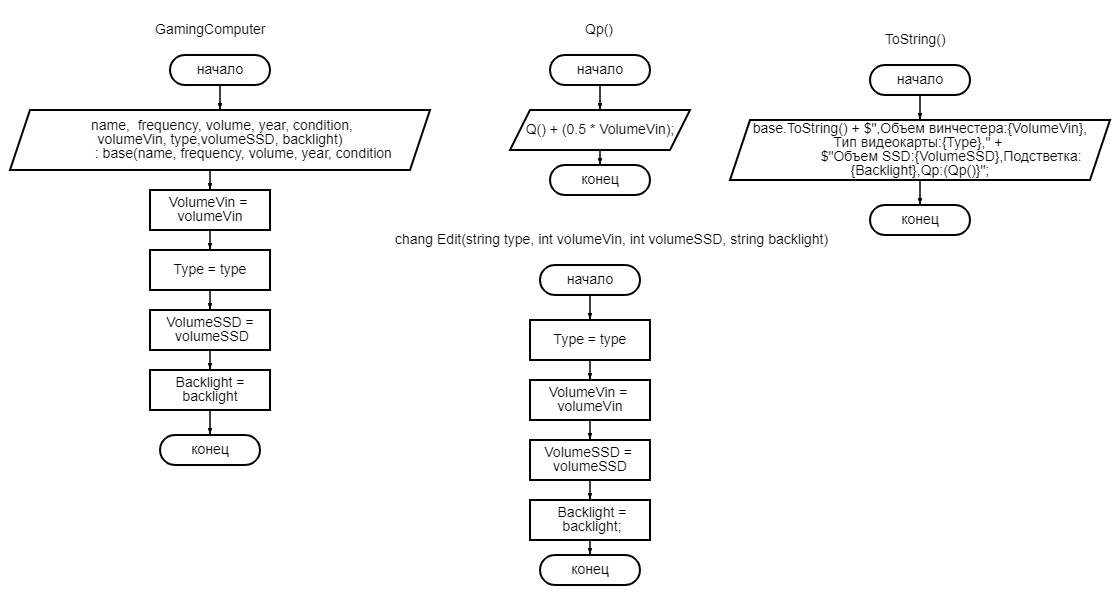
Метод Edit(string type, int volumeVin, int volumeSSD, string backlight) для редактирования свойств игрового компьютера.

1.3 Алгоритм решения

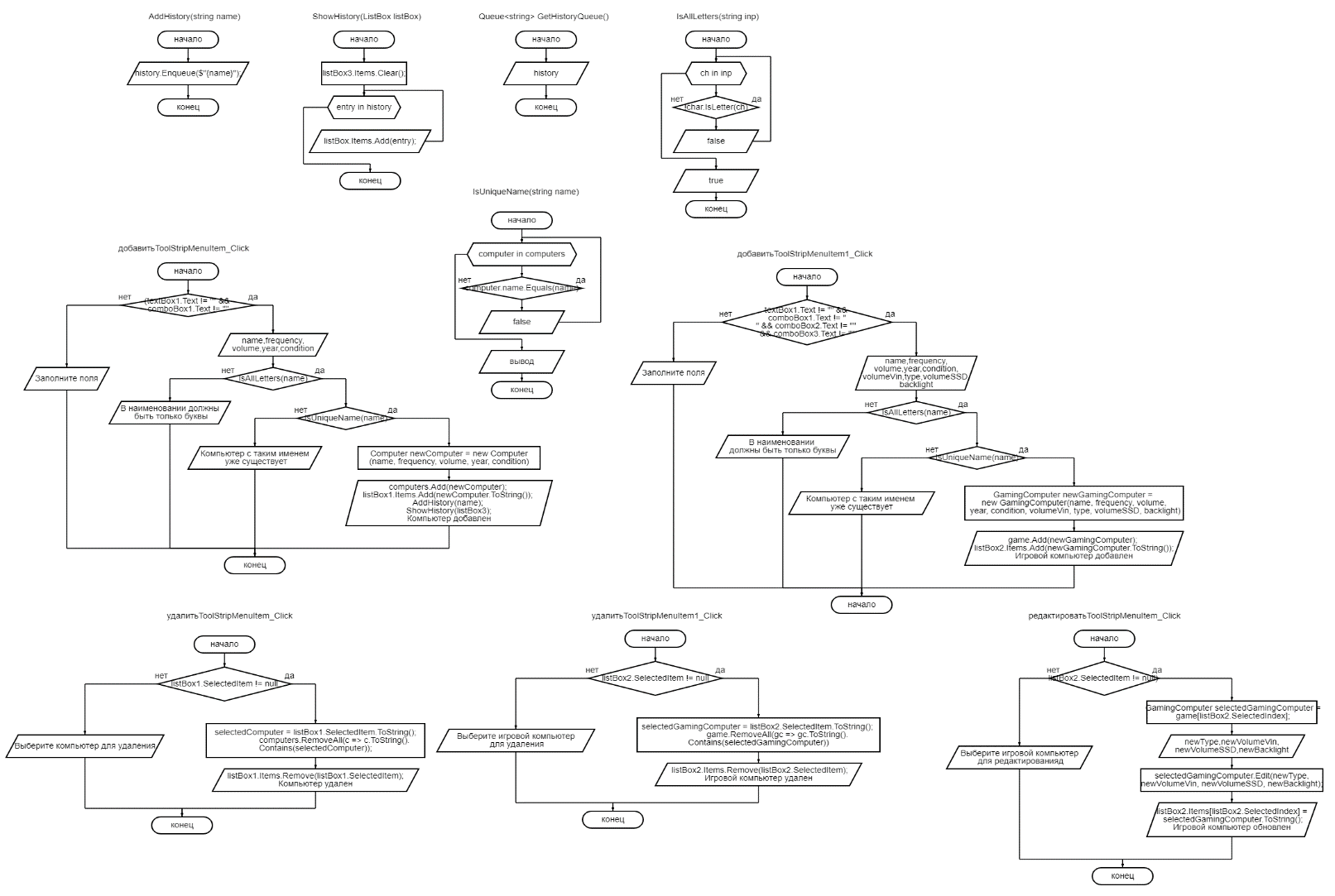
Блок схемы класса Computer:



Блок схемы класса GamingComputer:



Блок схемы класса Main:



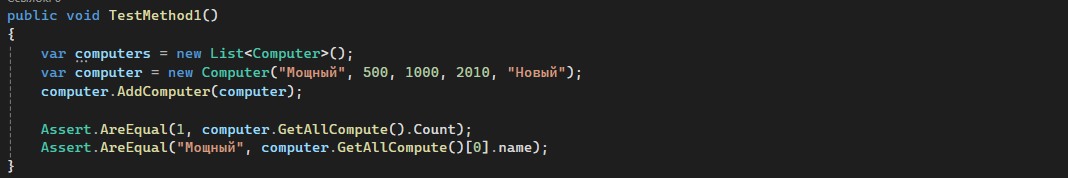
## 1.4 Используемые библиотеки

**System.Linq** – для работы с Linq.

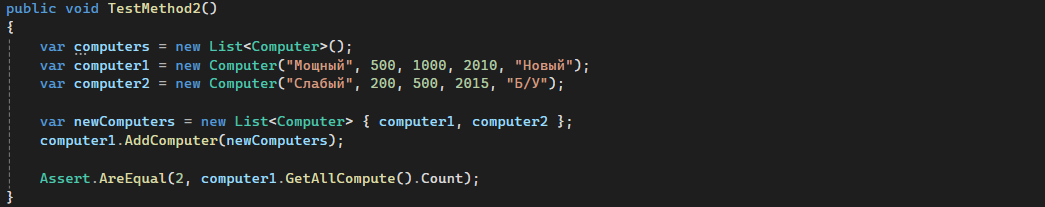
**System.Collections.Generic** – для работы с коллекциями.

**System.Windows.Forms** -для работы с элементами WindowsForms.

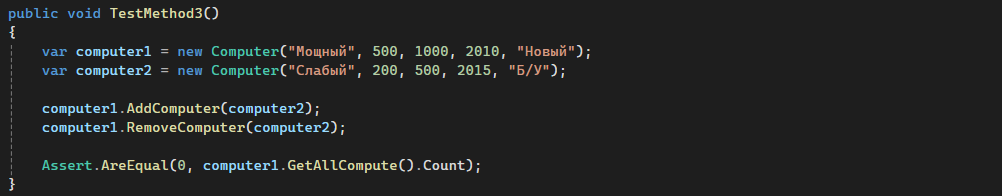
## 1.5 Тестовые случаи



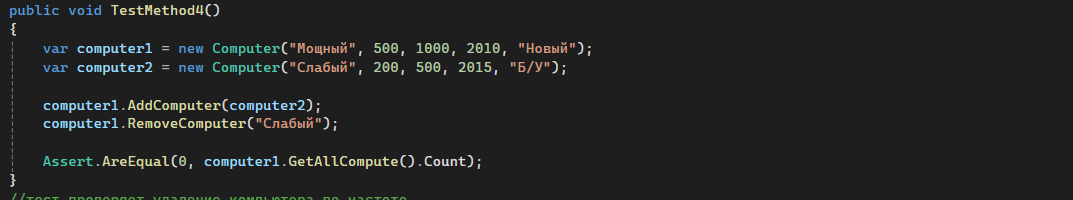
Тест проверяет добавление одного компьютера в список.



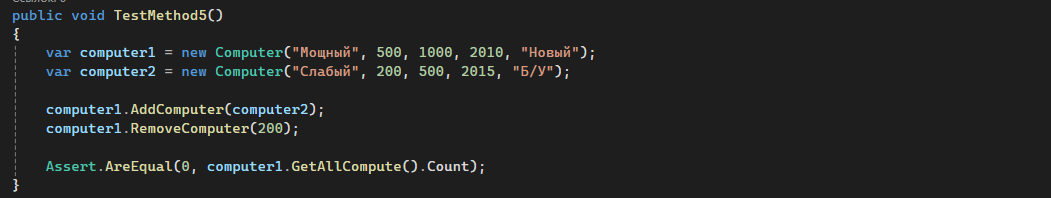
Тест проверяет можно ли заносить несколько компьютеров в список.



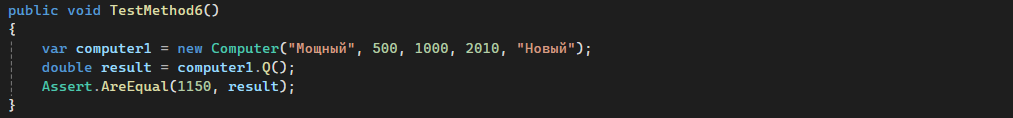
Тест проверяет удаление компьютера по объекту.



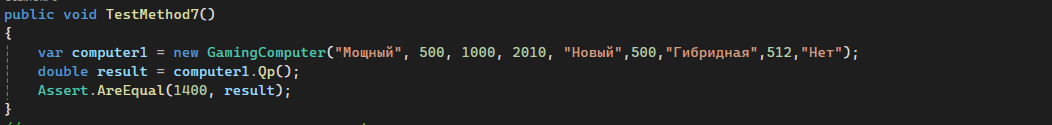
Тест проверяет удаление компьютера по имени.



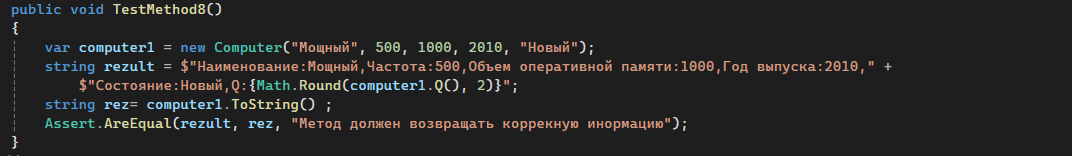
Тест проверяет удаление компьютера по частоте.



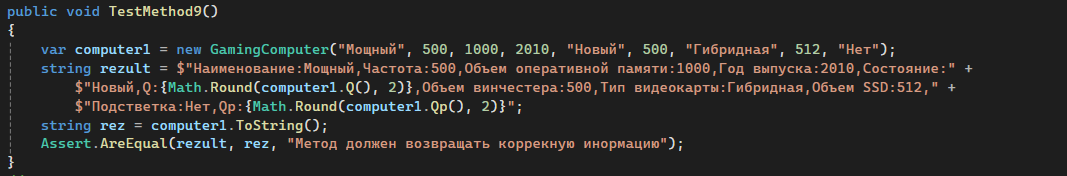
Тест проверяет, что метод Q() вычисляет значение на основе предоставленной формулы.



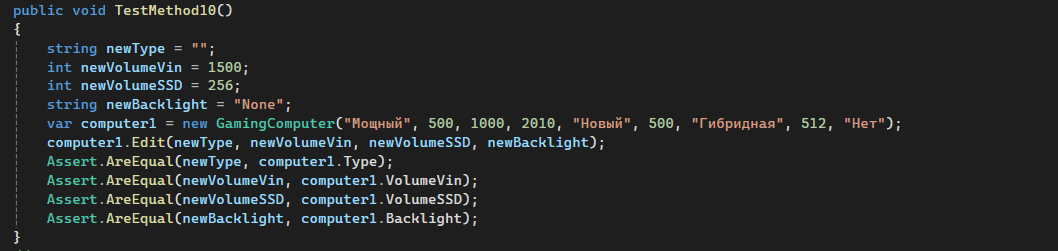
Тест проверяет, что метод Qр() вычисляет значение на основе предоставленной формулы.



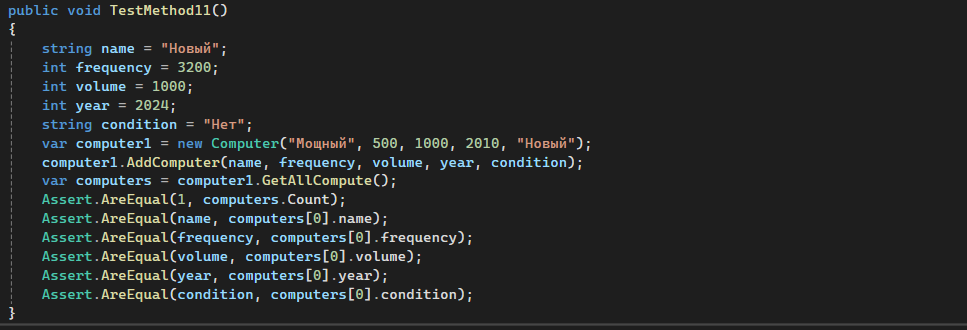
Метод для проверки корректности ввода информации о компьютере.



Тест для проверки корректности ввода информации о игровом компьютере.



Тест для проверки корректности редактирования информации.



Тест проверяет добавлен ли в список компьютер, и проверяет корректность данных.

## 1.6 Используемые инструменты

## Язык написания данной программы- C#

## Среда разработки – WindowsForms C#

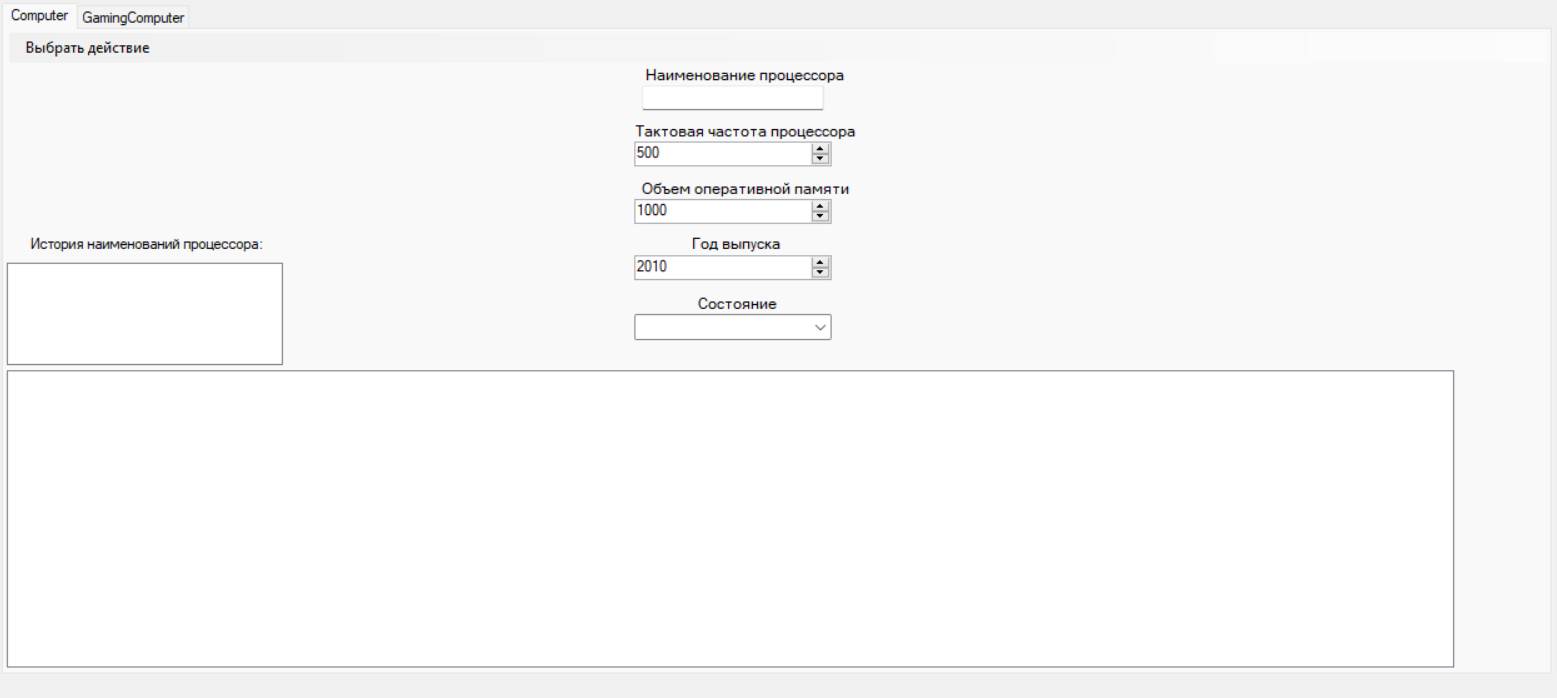
## 1.7 Описание пользовательского интерфейса

При запуске программы перед нами предстает экран, на который я добавила tabControl, в нем у меня есть tabPage1-Computer, на нем я также разместила 2 listBox, один для вывода информации о компьютере, второй для истории наименования процессора. Также я добавила textBox - наименование, numericUpDown – тактовая частота, объем оперативной памяти и год выпуска, comboBox – состояние, menuStrip-выбор действий: «добавление, удаление», при нажатии на эти кнопки, будут происходить действия в соответствии с их названиями.

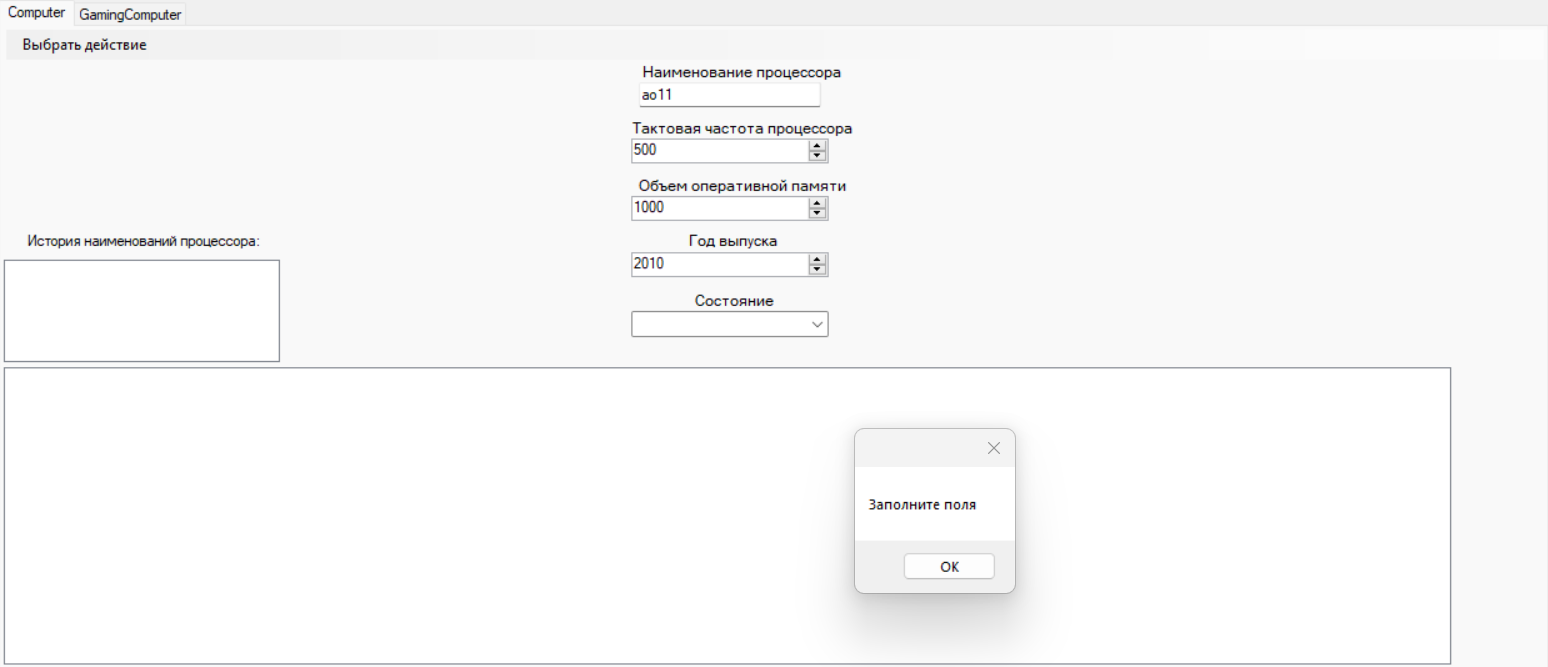
tabPage1-GamingComputer, на нем listBox для вывода информации о игровом компьютере, numericUpDown - объём винчестера и объем SSD, comboBox - тип видеокарты, тип видеокарты, menuStrip-выбор действий: «добавление, удаление, редактирование», », при нажатии на эти кнопки, также будут происходить действия в соответствии с их названиями. Также на этот экран выводится информация с, так tabPage1-Computer, как он является базовым классом.

## 1.8 Приложение (pr screen экранов)

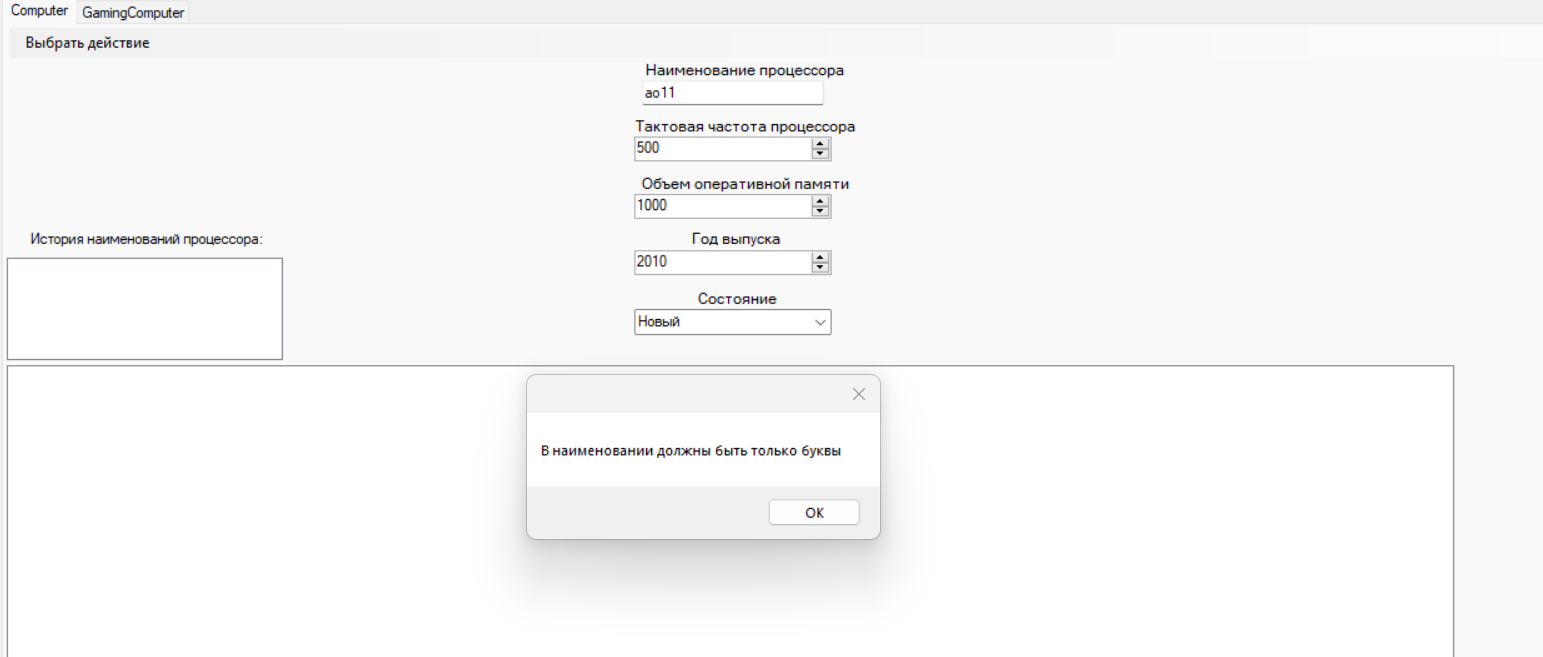
Computer:



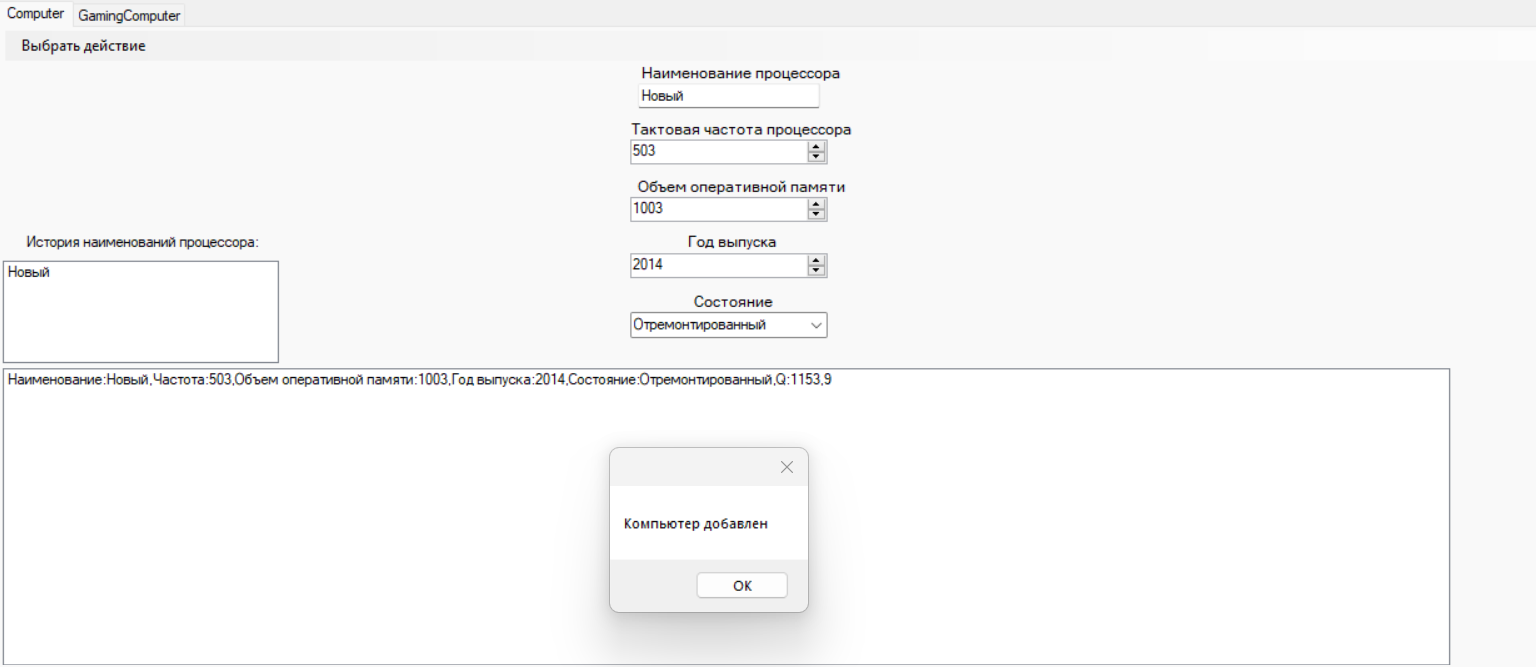
Главный экран



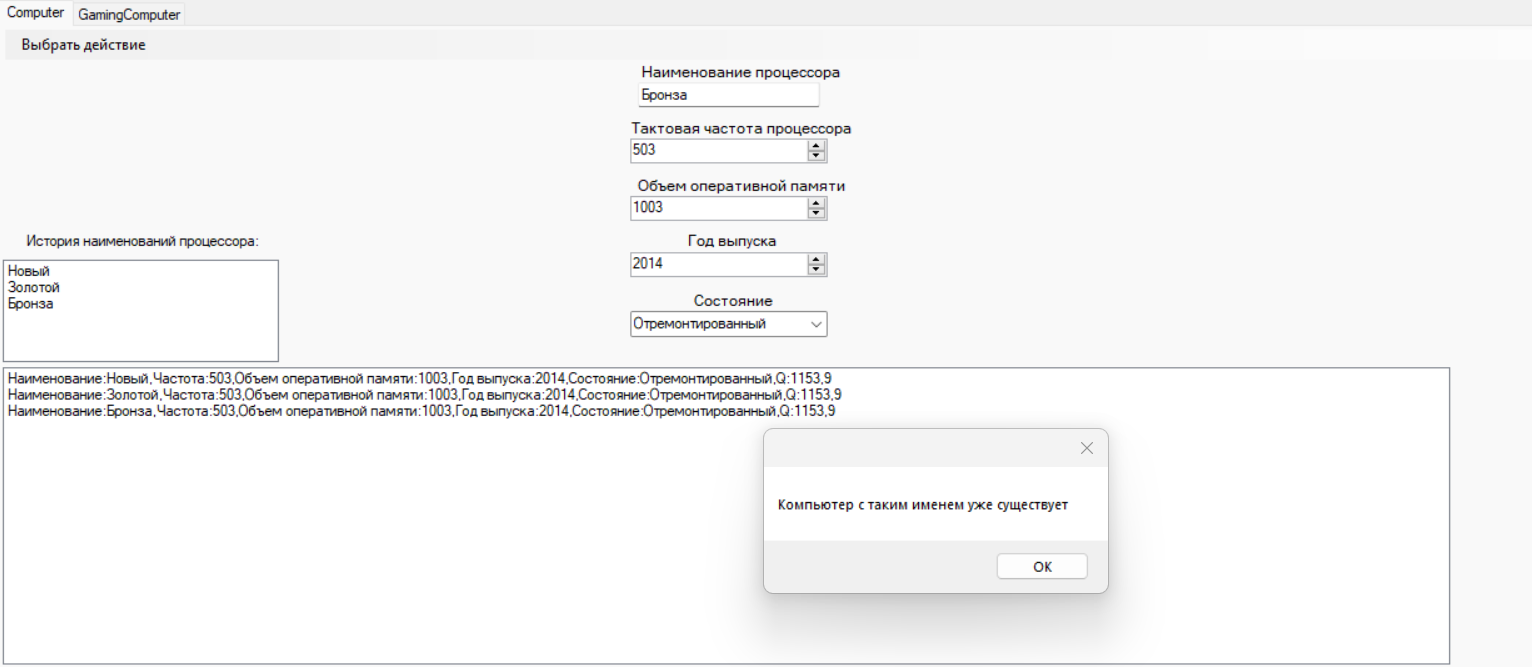
Проверка на пустые поля



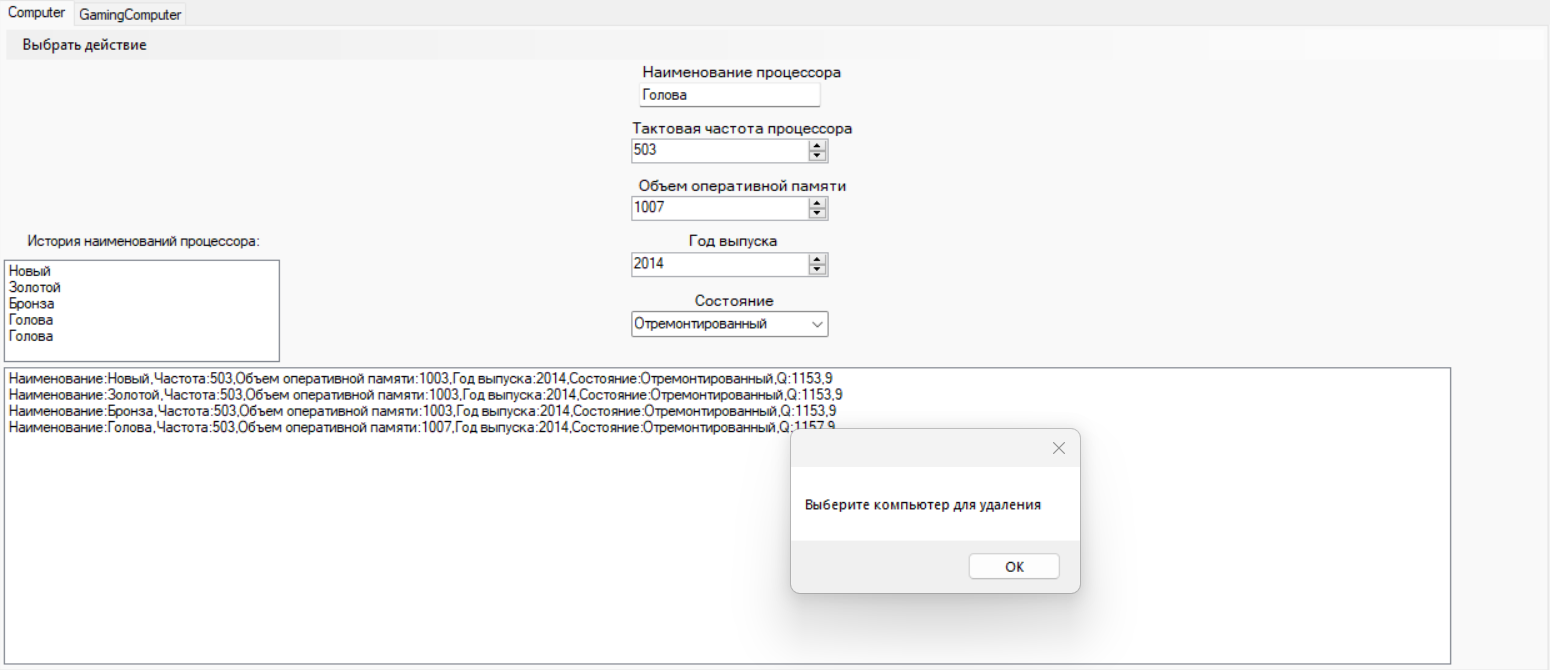
Проверка на корректный ввод наименования



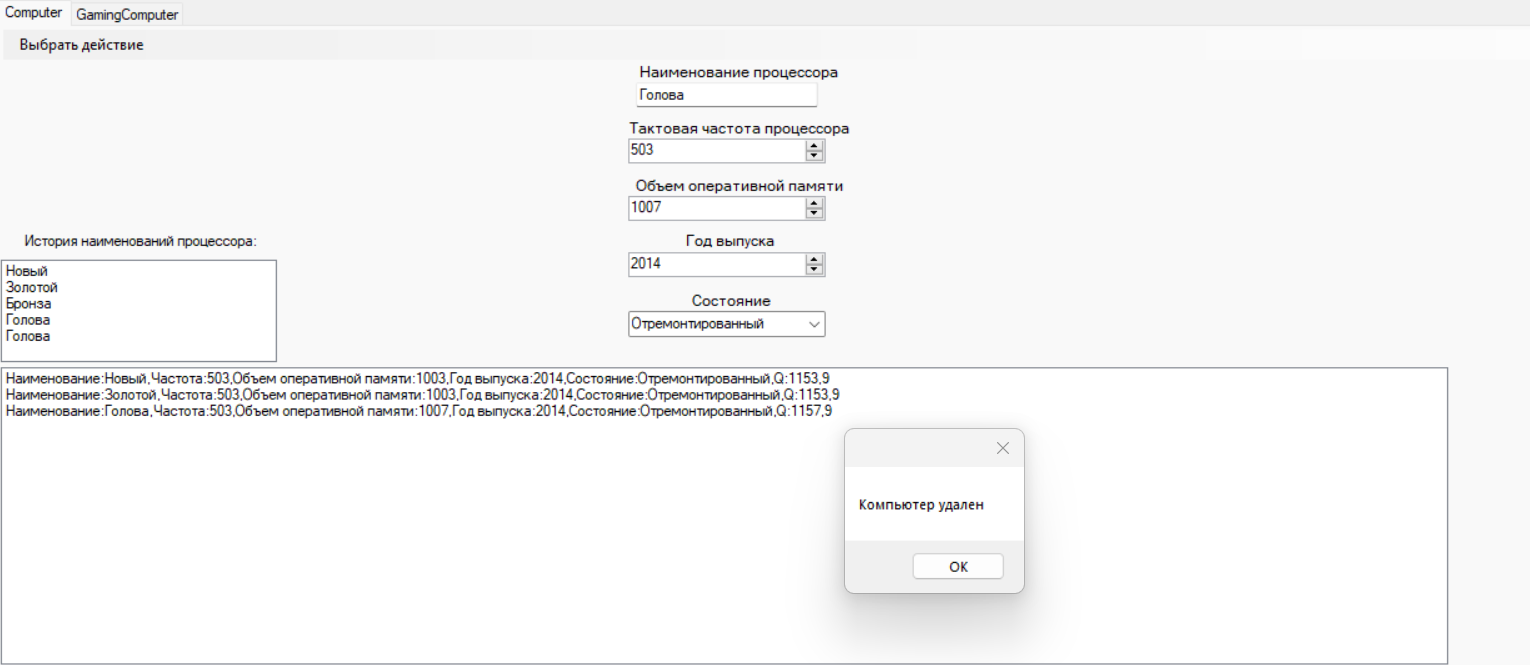
Добавление корректной информации



Проверка на уникальность

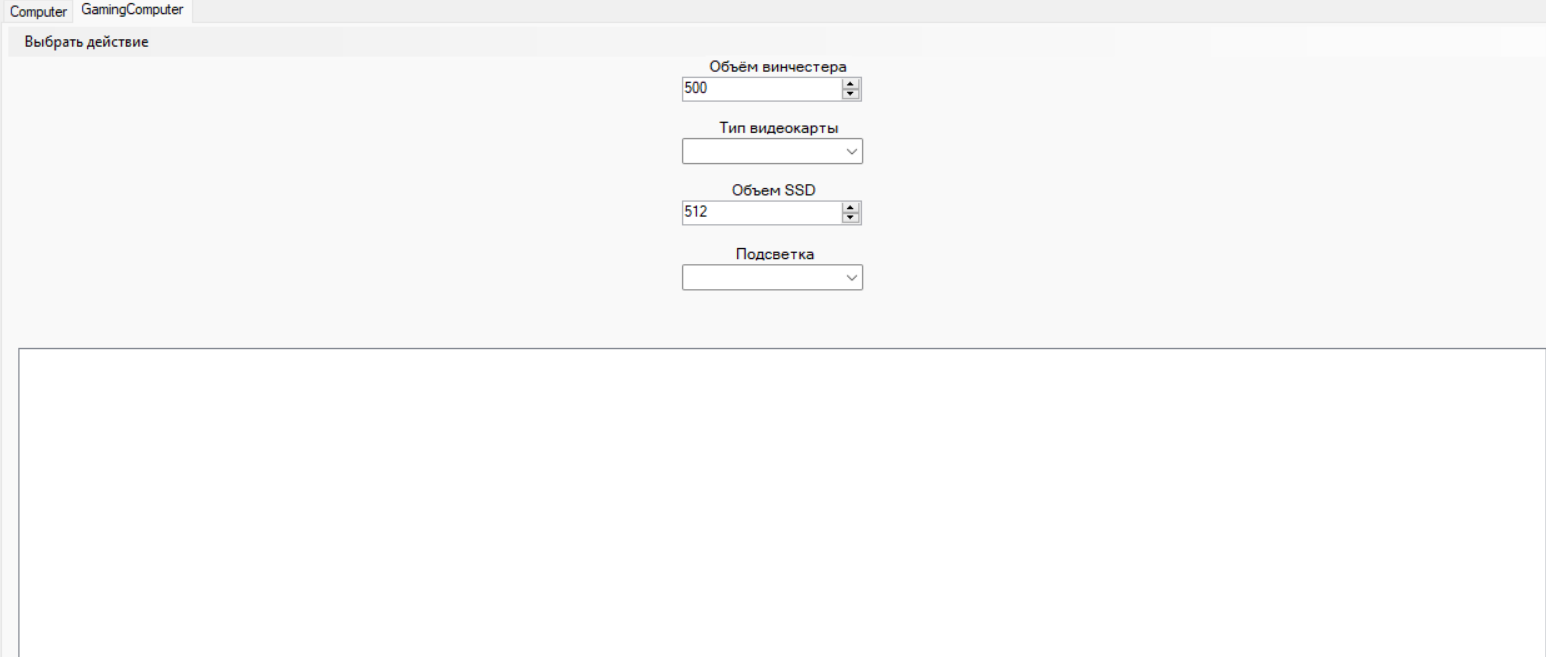


Проверка на выбор данных для удаления

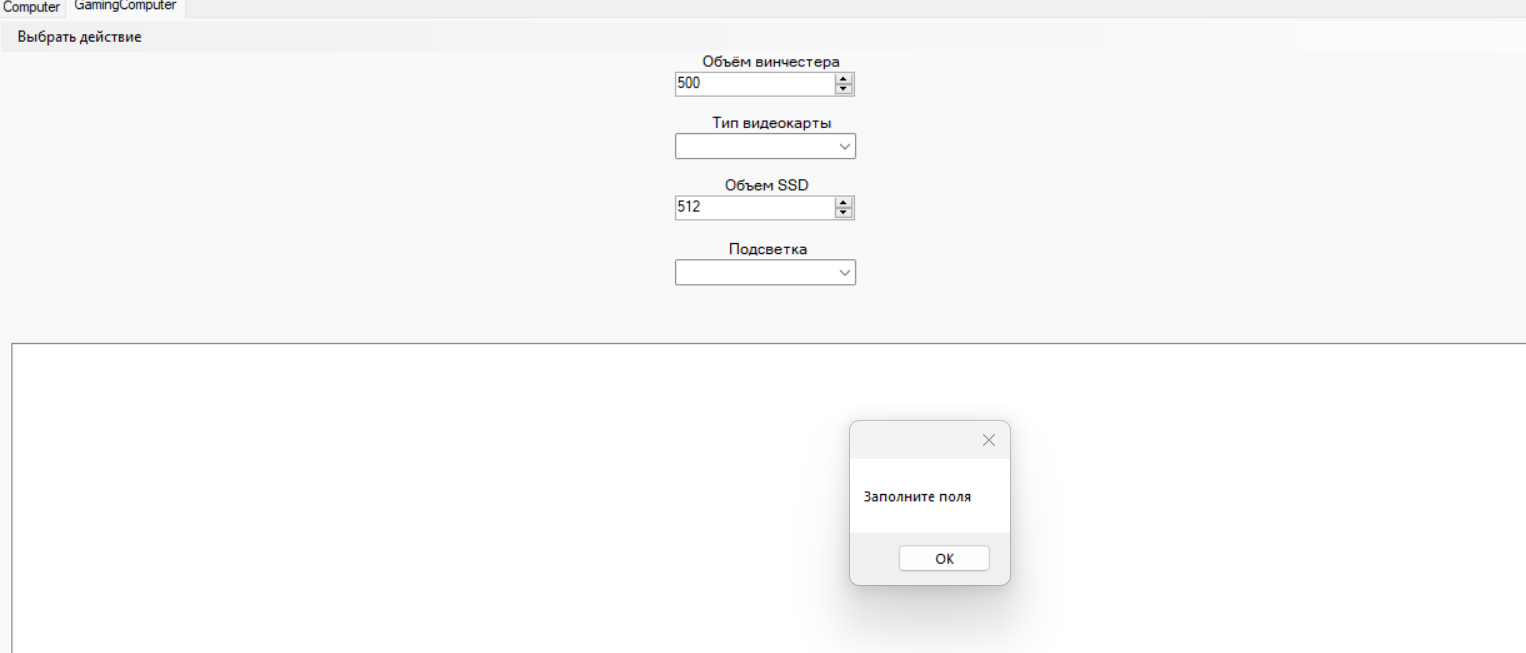


Удаление компьютера

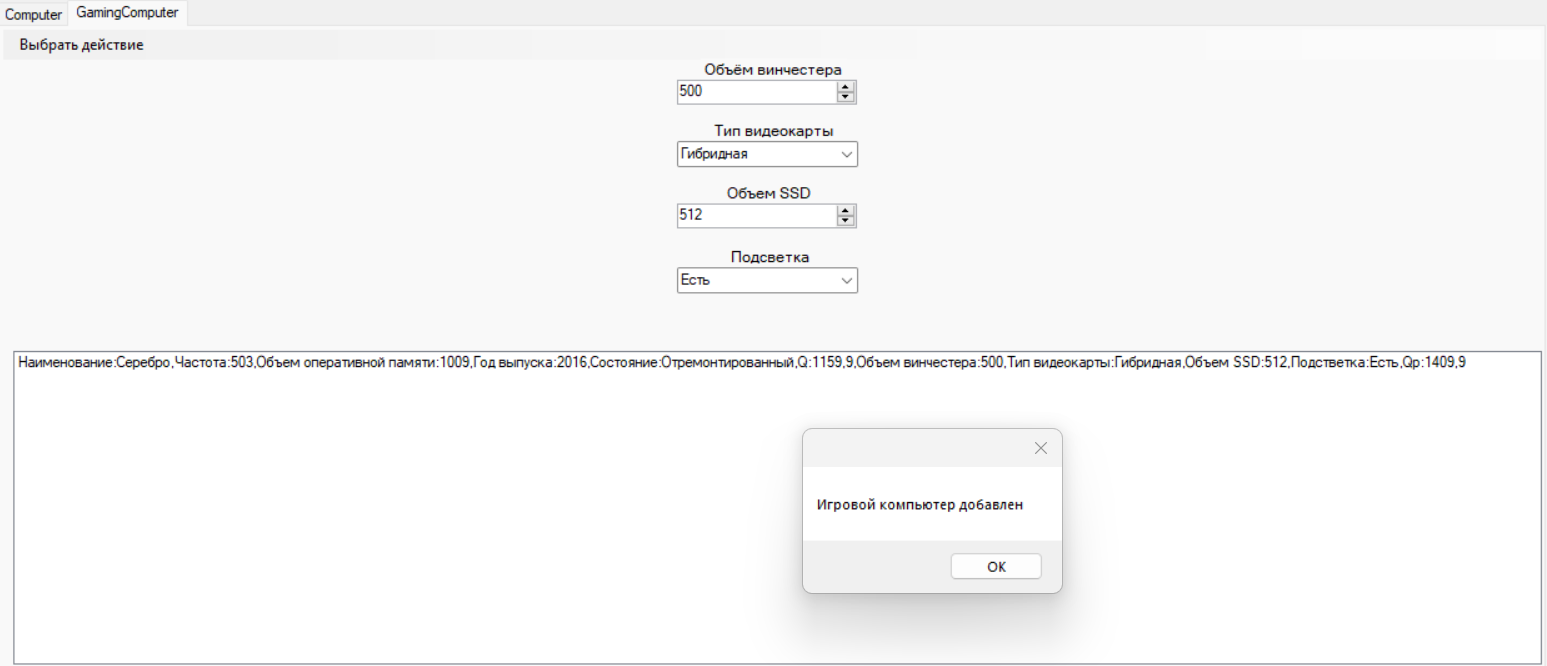
GamingComputer:



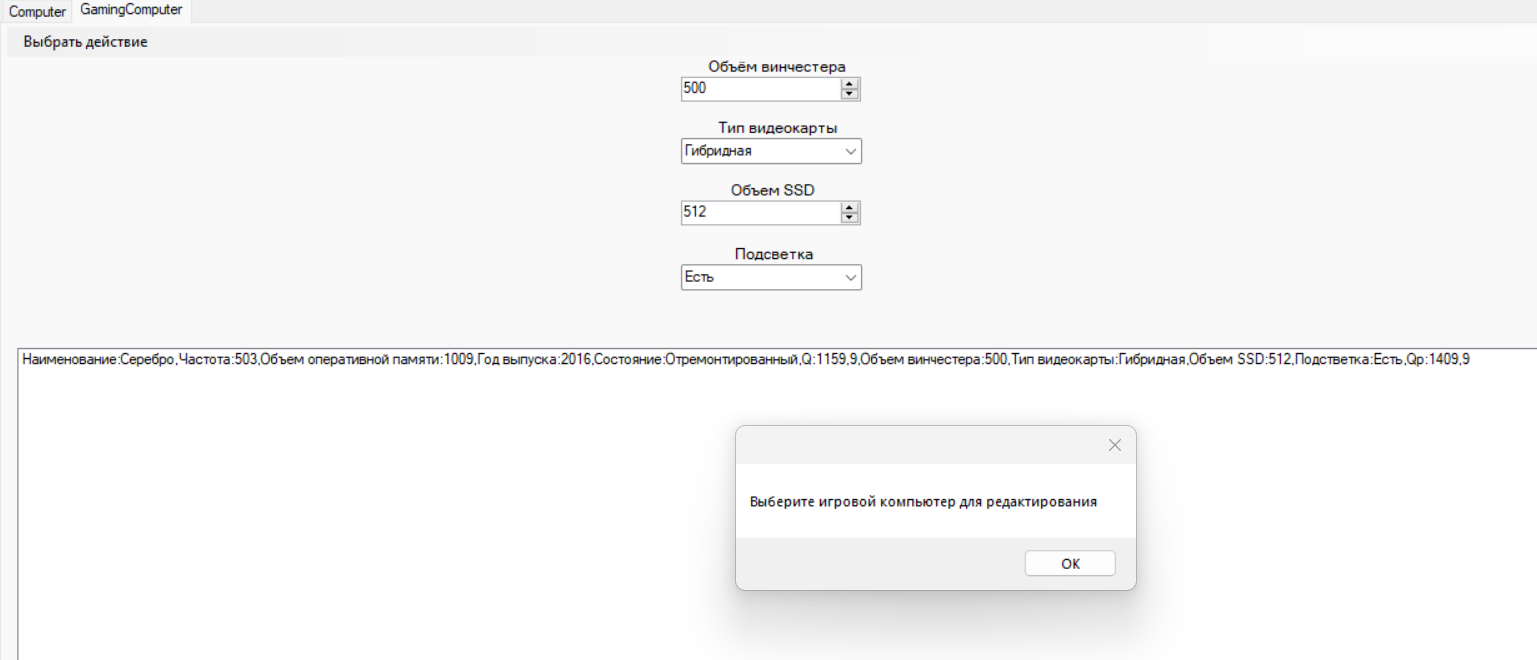
Главный экран



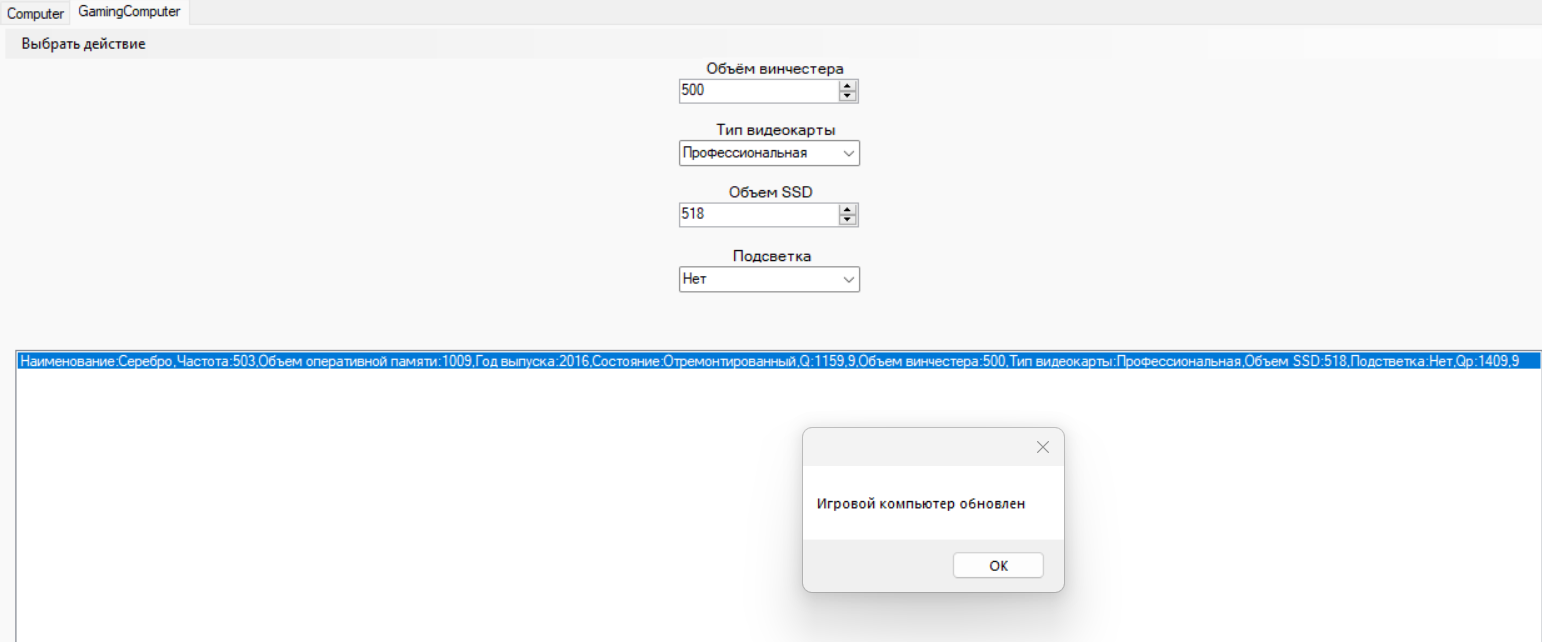
Проверка на пустые поля



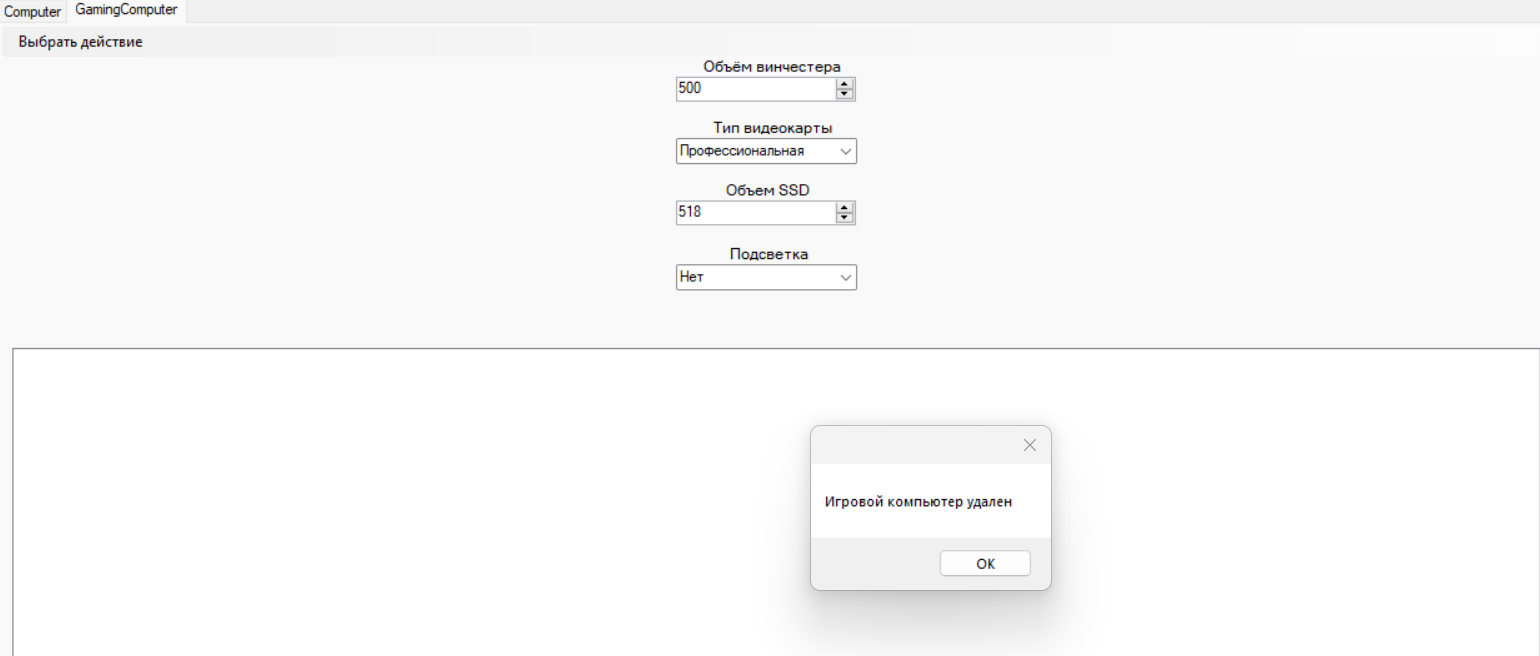
Добавление корректной информации



Проверка на выбор данных для редактирования



Редактирование игрового компьютера



Удаление игрового компьютера