Государственное бюджетное общеобразовательное учреждение города Москвы "Школа № 1501"

**ПРОЕКТ ПО ИНФОРМАТИКЕ**

**Разработка программы для проведения контроля знаний с помощью тестирования**

***Работу выполнил:***

Бежнар Алексей Сергеевич

ученик 11-4 класса

ГБОУ Школа № 1501

***Руководитель:***

ведущий программист

Бежнар Сергей Иванович

г. Москва

2019

Оглавление

[**Введение** 3](#_Toc29850620)

[**Требования к разрабатываемой программе** 5](#_Toc29850621)

[**Подготовка инструментов для создания программы** 6](#_Toc29850622)

[**Разработка программы** 8](#_Toc29850623)

[**Создание проекта** 8](#_Toc29850624)

[**Структура проекта** 10](#_Toc29850625)

[**Интерфейсы программы** 12](#_Toc29850626)

[**Заключение** 16](#_Toc29850627)

[**Список литературы и ресурсов в сети интернет** 17](#_Toc29850628)

# **Введение**

В настоящее время при обучении используются средства вычислительной техники, которые позволяют усовершенствовать процесс обучения, повысить его эффективность. Сегодня учитель должен следовать за развитием информационных технологий. Они диктуют новые формы и методы в педагогическом труде. Для выявления результатов обучения актуален контроль знаний учащихся, который можно осуществить в том числе с помощью тестирования с применением вычислительных средств.

**Проблема:**

В сети интернет существует множество образовательных ресурсов и программ, которые предлагаются в качестве дополнительных инструментов в образовательном процессе. Однако для конкретного урока учителю нужен определенный обучающий материал только для данного класса и урока, возможность проверки усвоения материала по уроку. Поэтому возникает необходимость в использовании простого и понятного инструмента, который позволит педагогу «настраивать» проверку знаний.

**Решение проблемы:**

Одним из инструментов для достижения цели является создание контролирующей программы, позволяющей создавать и проводить тестирование, учитывая потребности учителей и учащихся.

**Цель проекта:**

Создание контролирующей программы, позволяющей создавать и проводить тестирование с использованием вычислительной техники.

**Задачи проекта:**

* Обеспечить возможность создания тестов;
* Обеспечить возможность проведения тестирования на усвоение учебного материала, полученного на уроках.

Передо мной в рамках данного проекта стоят следующие задачи:

* закрепить знания, полученные при изучении предмета «Информатика»;
* получить опыт разработки небольшого кроссплатформенного приложения;
* изучить кроссплатформенную библиотеку пользовательского интерфейса AvaloniaUI.

**Ход проекта:**

* Осмысление цели проекта;
* Изучение литературы, интернет-ресурсов по проблематике вопроса;
* Разработка простого и удобного пользовательского интерфейса программы;
* Разработка программы на языке объектно-ориентированного программирования C#;
* Тестирование программы, исправление ошибок;
* Представление результата.

**Результат представления проекта:**

* Компьютерная программа для проведения тестирования;
* Информационные ресурсы: интернет-ресурсы с описанием и исходными кодами по проекту.

# **Требования к разрабатываемой программе**

При разработке и создании программы для проведения контроля знаний с помощью тестирования мной были сформулированы следующие требования, предъявляемые к программе:

* Простота и удобство работы с программой;
* Возможность создания и редактирования тестов;
* Возможность выбора и прохождения теста;
* Возможность просмотра результатов прохождения выбранного теста;
* Программа должна быть разработана с учетом возможности запуска на различных операционных системах – Windows, Linux, MacOS, т.е. быть кроссплатформенной.

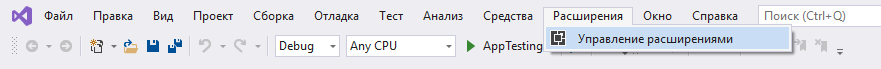
# **Подготовка инструментов для создания программы**

Для того, чтобы начать разрабатывать программу необходимо произвести ряд подготовительных действий:

1. Скачать инсталлятор среды разработки Visual Studio 2019 (на момент начала разработки самая последняя актуальная версия) и установить её.
2. Установить расширение для Visual Studio 2019 «Avalonia for Visual Studio».

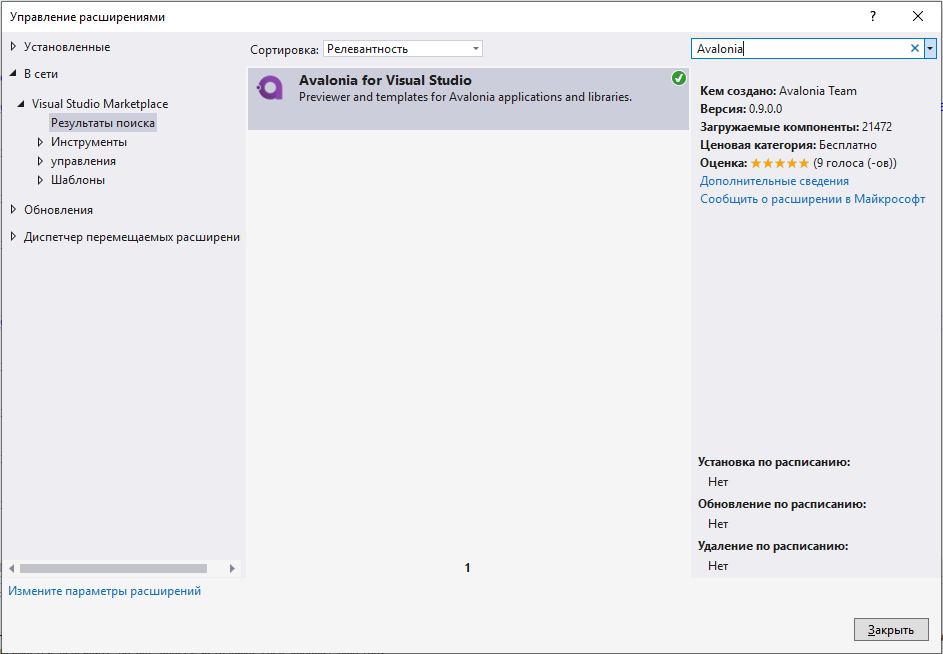
Скачивание инсталлятора среды разработки Visual Studio 2019 производилось с сайта <https://visualstudio.microsoft.com/vs/>. Установка Visual Studio 2019 с помощью инсталлятора имеет интуитивно понятный интерфейс, поэтому процесс установки не требует пояснений.

После установки Visual Studio 2019 для разработки кроссплатформенной программы с пользовательским интерфейсом необходимо установить расширение «Avalonia for Visual Studio». Для этого необходимо в Visual Studio 2019 открыть пункт меню «Расширения» и в появившемся списке выбрать пункт «Управление расширениями»:



**Рис.1 Выбор пункта меню**

В появившемся диалоге «Управление расширениями» в строке поиска набрать слово Avalonia. По этому слову будет произведен поиск расширений, содержащих это слово, в отфильтрованном списке найти расширение «Avalonia for Visual Studio», выделить его и нажать кнопку «Скачать»:



**Рис.2 Поиск расширения**

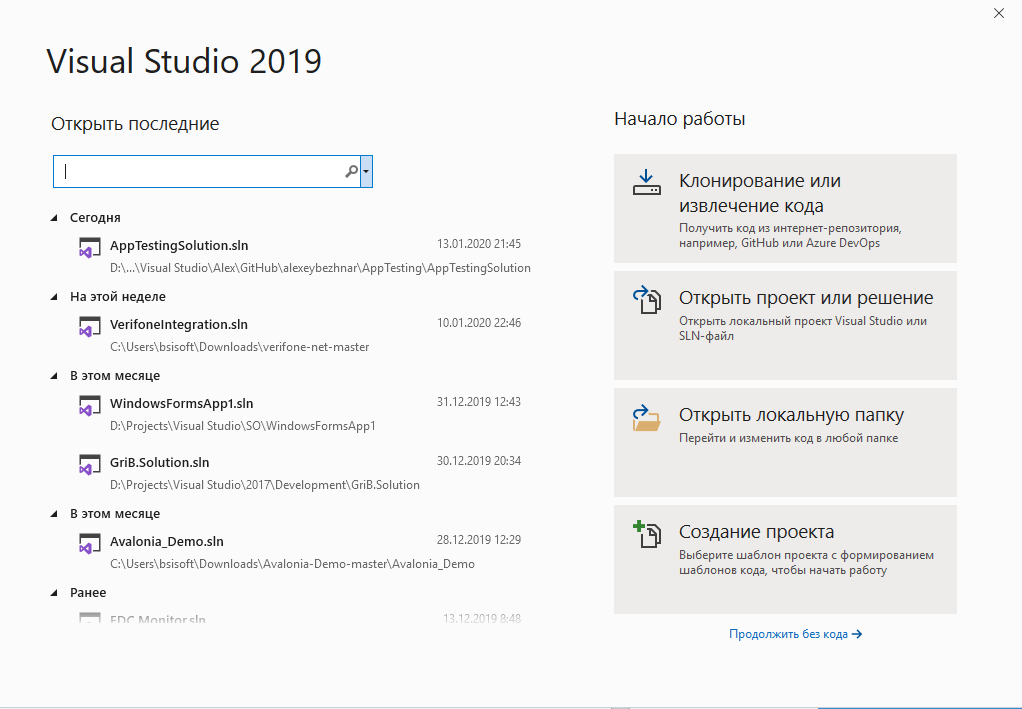
Далее необходимо дождаться завершения скачивания и перезапустить Visual Studio 2019, чтобы завершить установку расширения.

После выполнения перечисленных выше действий подготовка инструментов для начала разработки программы завершена.

# **Разработка программы**

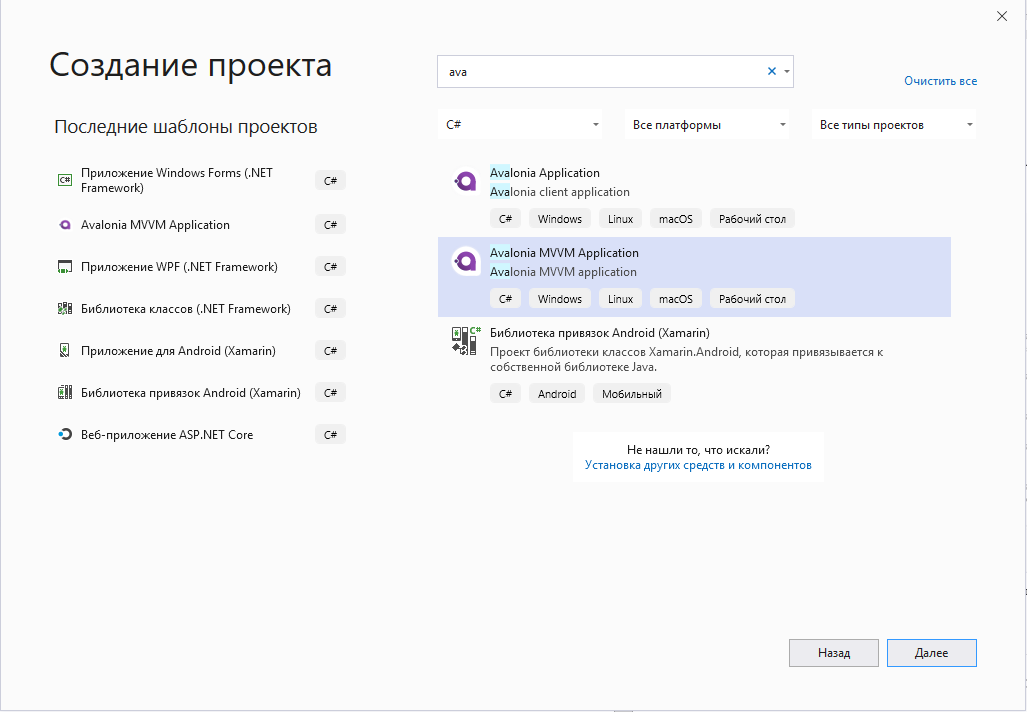
# **Создание проекта**

При открытии среды разработки Visual Studio 2019 открывается диалоговое окно начала работы, в котором необходимо выбрать пункт «Создание проекта»:



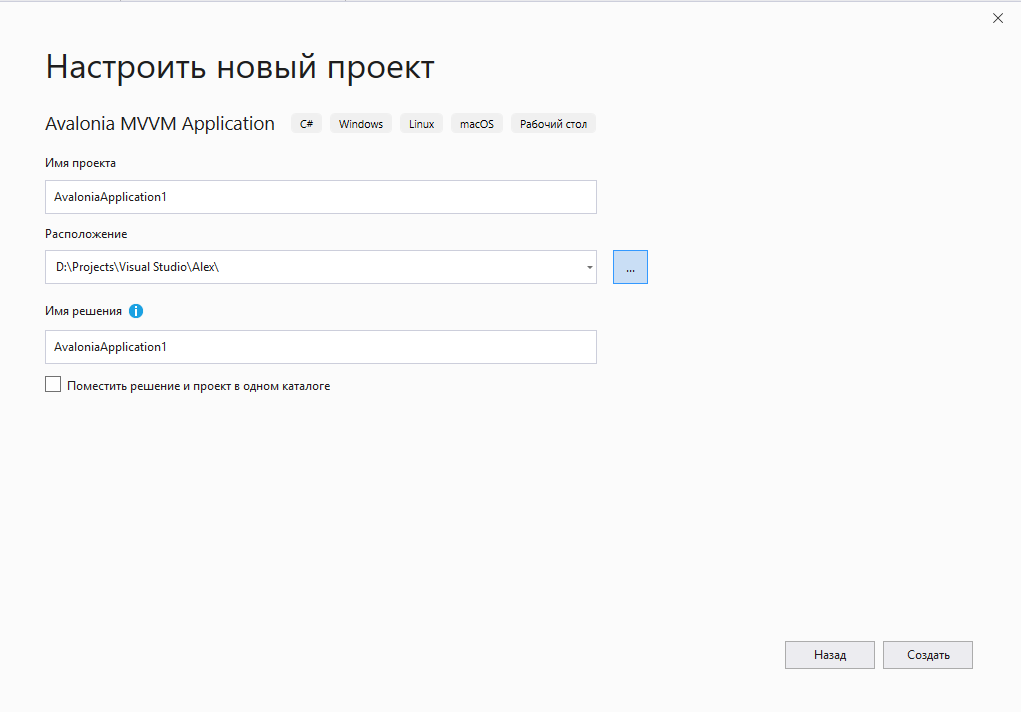
**Рис.3 Выбор пункта «Создание проекта»**

После выбора пункта «Создание проекта» откроется диалоговое окно «Создание проекта», в котором необходимо найти требуемый шаблон проекта для создания (в нашем случае это «Avalonia MVVM Application»), выделить его и нажать кнопку «Далее»:



**Рис.4 Выбор шаблона проекта**

После выбора шаблона проекта и нажатии на кнопку «Далее» будет открыто диалоговое окно «Настроить новый проект», в котором необходимо указать имя проекта, расположение в файловой системе и название решения:

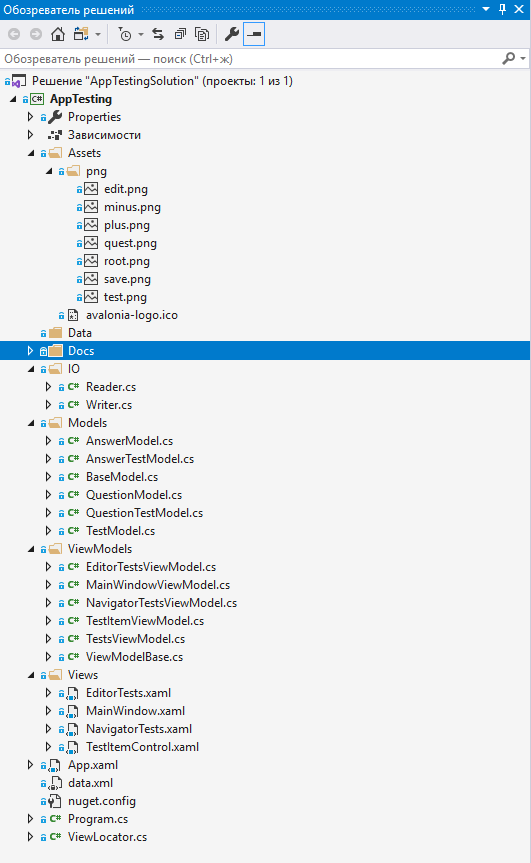


**Рис.5 Задание имен при создании проекта**

После нажатия на кнопку «Создать» будет создана заготовка решения с преднастроенным проектом для разработки.

# **Структура проекта**

Структура проекта, которая была создана при разработке программы для проведения тестирования представлена на рисунке ниже:



**Рис.6 Структура проекта**

Назначение папок:

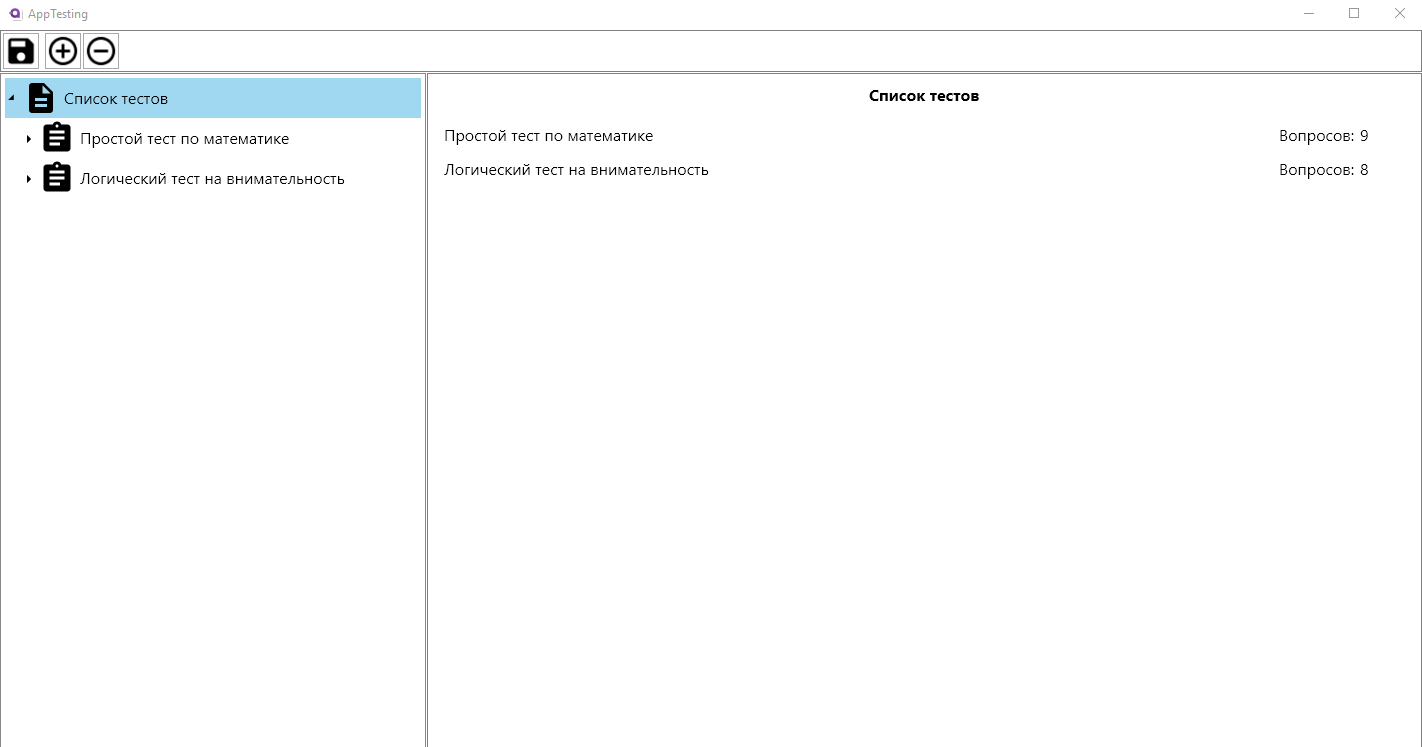
* Папка «Assets» – ресурсы программы в виде картинок и логотипа программы;
* Папка «Docs» – содержит презентацию и документ описания проекта;
* Папка «IO» – исходный код вспомогательных классов для чтения и сохранения тестов;
* Папка «Models» – исходный код классов для описания структур данных программы;
* Папка «ViewModels» – исходный код классов для взаимодействия представлений (форм, диалоговых окон, элементов управления и контролов) с пользователем и управления данными;
* Папка «Views» – исходный код и разметка на языке XAML форм, диалоговых окон, элементов управления и контролов для взаимодействия с пользователем и управления данными;
* Файл «App.xaml» – файл разметки для подключения ресурсов приложения и исходный код класса нашего приложения;
* Файл «Program.cs» – файл с исходным кодом «точки входа» (старта) приложения.

# **Интерфейсы программы**

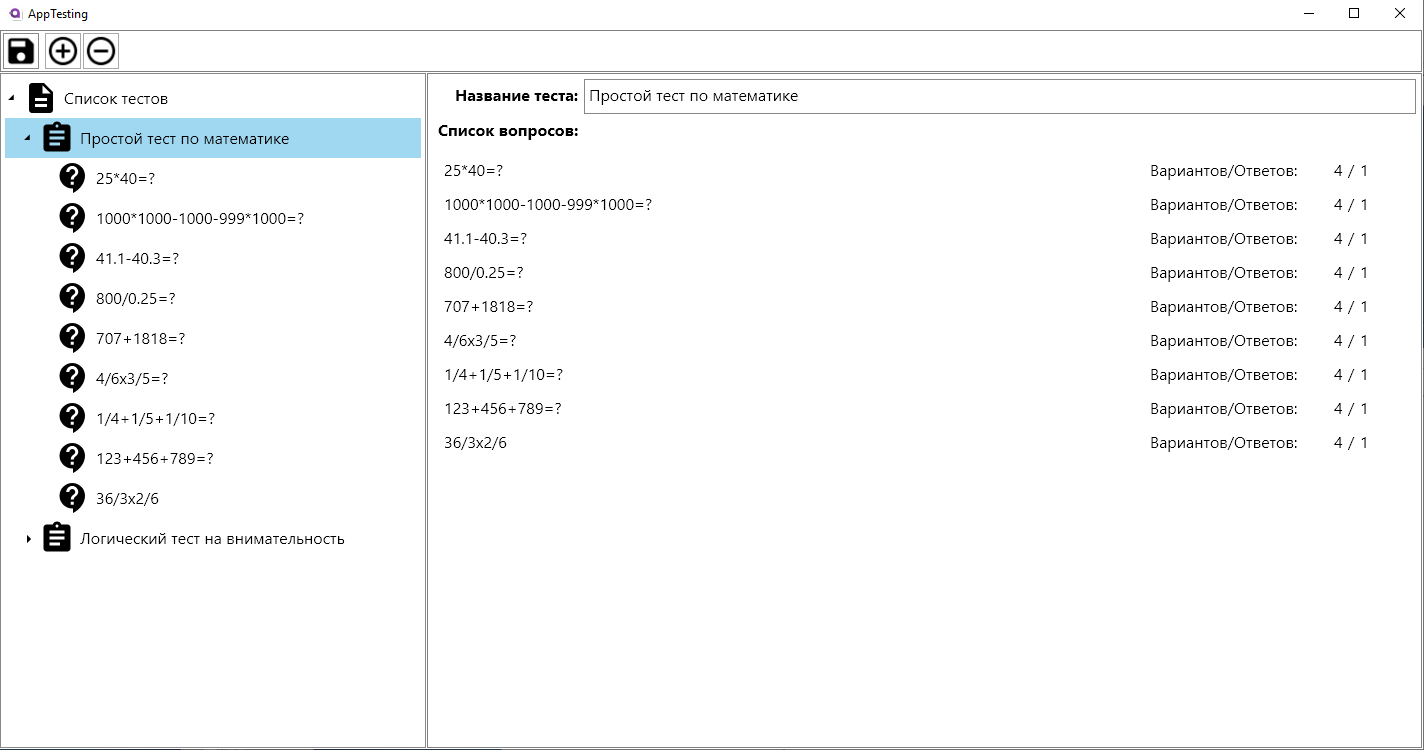
Программа логически разделена на 2 интерфейса:

* Интерфейс администратора – доступен при запуске программы с ключом командной строки «/admin»;
* Интерфейс пользователя.

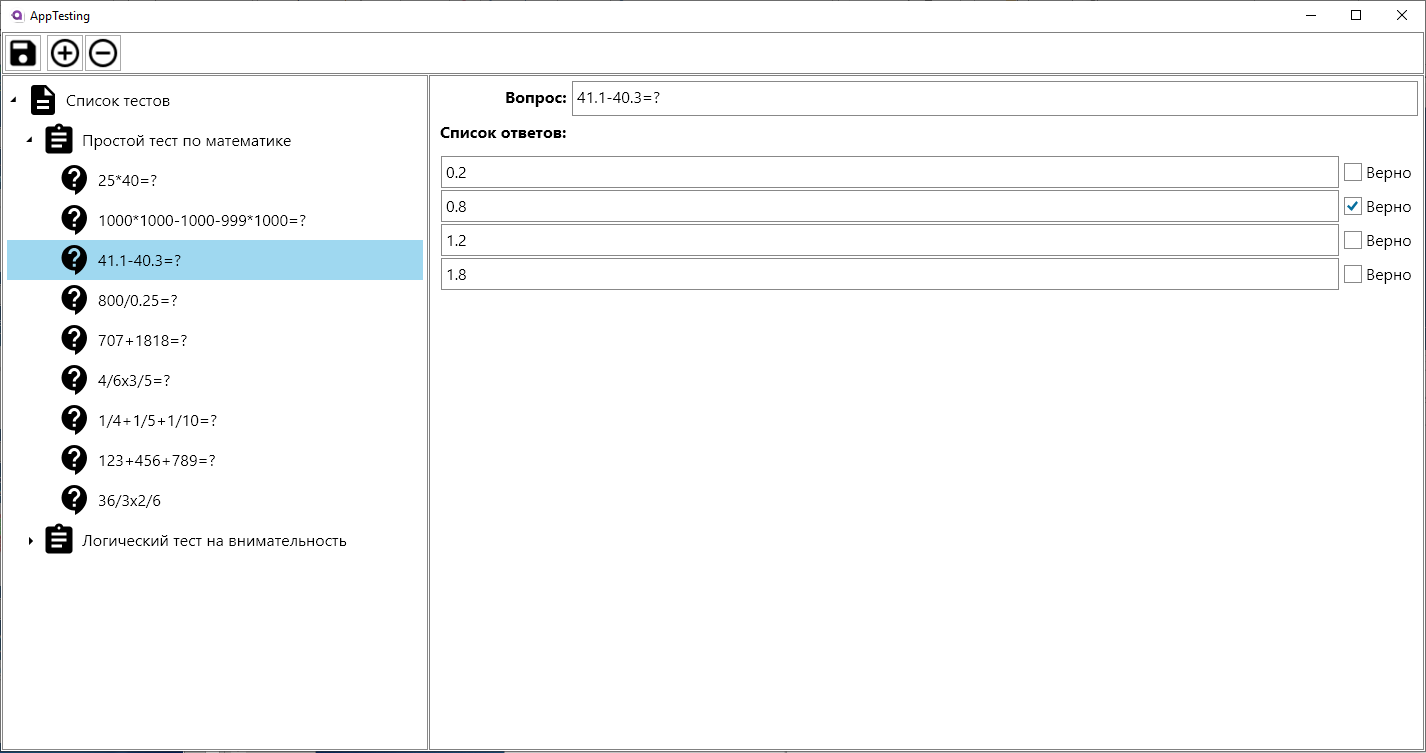
Интерфейс администратора представляет собой редактор, в котором производится редактирование списка тестов, списка вопросов теста, варианты ответов к каждому вопросу и задание правильных ответов к вопросу. Внешний вид представлений для редактирования тестов представлен на следующих рисунках ниже:



**Рис.7 Программа, запущенна в режиме «Администратор»**

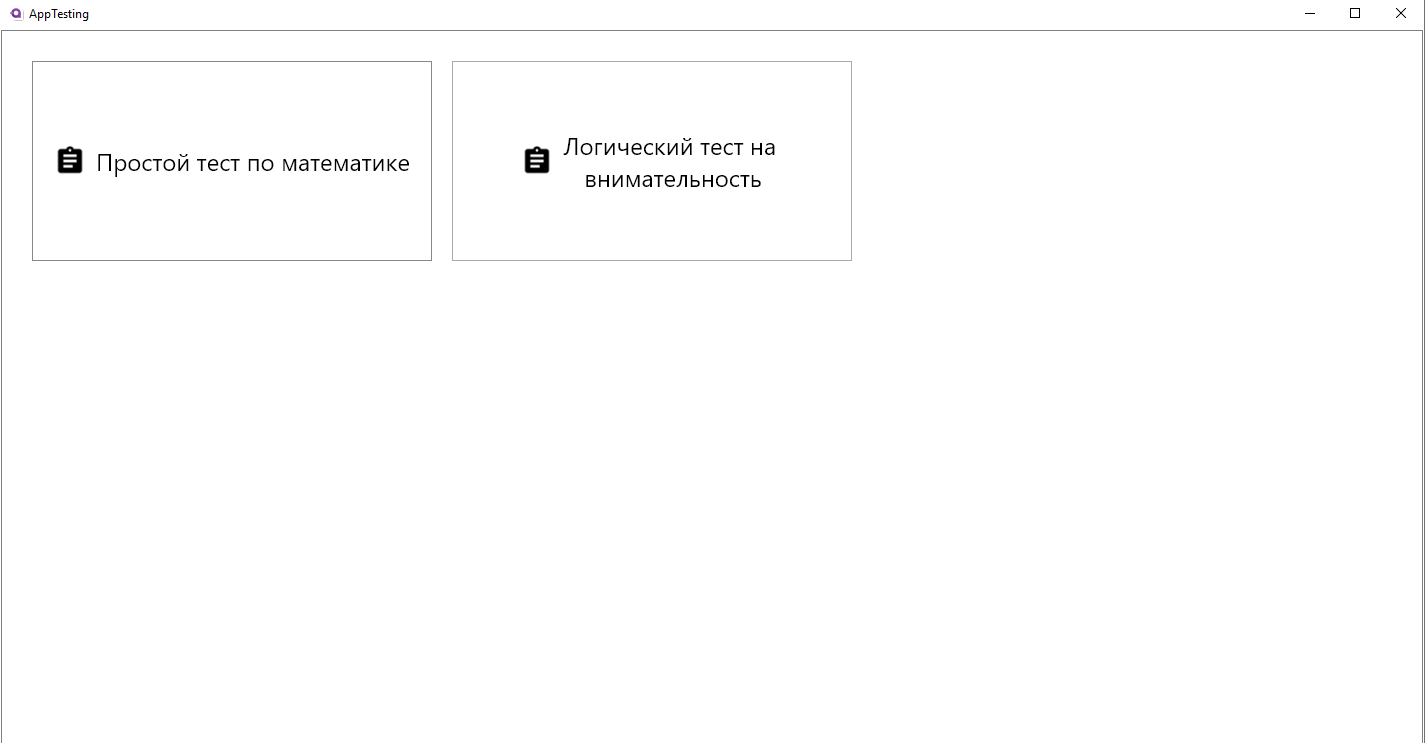


**Рис.8 Просмотр и редактирование теста**

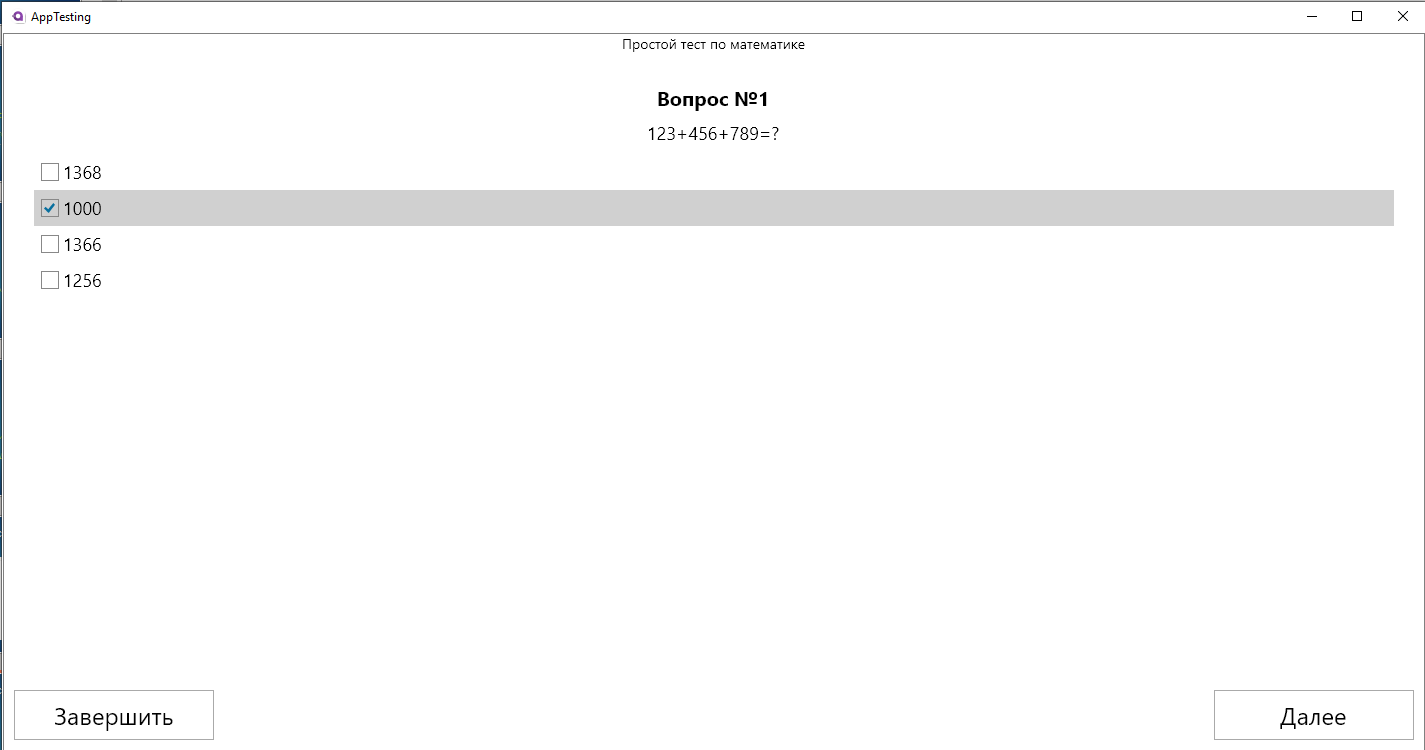


**Рис.9 Просмотр и редактирование вопроса теста и ответов на него**

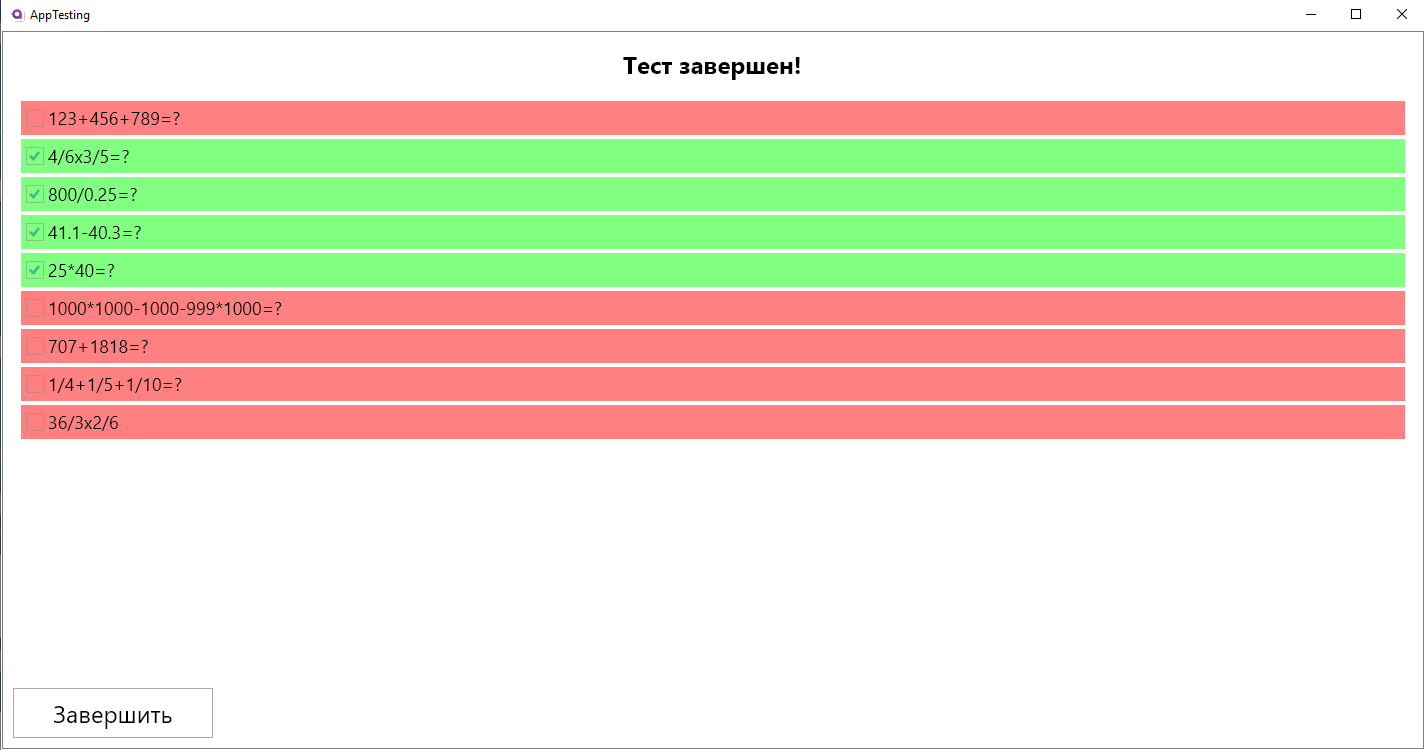
Интерфейс пользователя представляет собой навигатор по списку тестов. Пользователь может выбрать из имеющихся тестов необходимый и выполнить его, отвечая на вопросы теста. В конце прохождения теста пользователю будет показан результат выполнения теста. Внешний вид представлений для пользователя представлен на следующих рисунках ниже:



**Рис.10 Главное окно программы с тестами**



**Рис.11 Окно с вопросом теста и вариантами ответов**



**Рис.12 Окно с результатами прохождения теста**

# **Заключение**

При разработке и создании программы для проведения контроля знаний с помощью тестирования мной был получены следующие результаты:

* Получен опыт программирования в среде разработки Visual Studio 2019 с использованием языка программирования C#;
* Апробирована кроссплатформенная технология создания приложений с использованием библиотеки пользовательского интерфейса AvaloniaUI;
* Создано кроссплатформенное приложение для проведения контроля знаний с помощью тестирования;
* Получен навык работы с системой управления версиями github. Результаты работы выложены по адресу: <https://github.com/alexeybezhnar/AppTesting>
* Выполнены требования, указанные в разделе «Требования к разрабатываемой программе».

# **Список литературы и ресурсов в сети интернет**

1. <https://visualstudio.microsoft.com/vs/>. Сайт среды разработки Visual Studio.
2. <https://avaloniaui.net/>. Сайт, содержащий примеры и документацию кроссплатформенной библиотеки Avalonia UI.
3. <https://github.com/alexeybezhnar/AppTesting>. Разрабатываемые решение и проект, выложенные в систему управления версиями GITHUB.