[Агаджанян Давид 10-2

Школа №33]

Разработка программы для проверки знаний учащихся

Тема:

Многим учителям не хватает бумаги а также сил проверять большое количество тестов и работ учеников

Цель:

Создать программу которая автоматизирует проверку и сократит затрату на бумаги.

Возможность:

-создание тестов

-проверка тестов

-сохранение тестов в txt формат для дальнейшего использования

Задачи:

1. Исследовать аналогичные программы
2. Выявить функционал который я буду разрабатывать
3. Изучить работу библиотеки PyQt6(5)
4. Создать программу
5. Протестировать её
6. Проанализировать результаты и представить готовую модель

Глава 1 - Исследование

1.1: Анализ прототипов

Я нашёл прототипы моей программы такие как:

-Net Test(1)

-mytestxpro(2)

-test pad online(3)

|  |  |  |  |
| --- | --- | --- | --- |
| Названия программы | Net Test | mytestxpro | test pad online |
| Стоимость либо наличие платного контента | бесплатная | Есть платный контент | бесплатная |
| Доступность(нужна ли подключаться к интернету | нет | да | да |
| Понятный интерфейс | + | нет | да |
| Дизайн | - | - | + |

Вывод: Учитывая функционал данных программ, дизайн, платный контент и их доступность. Я хочу создать свою программу которая будет иметь хороший функционал с полностью бесплатным контентом и она будет работать offline.

1.2 Технологии:

-Для реализации задачи я буду использовать язык программирования Python(4)

-Использую в своей программе PyQt6(5)(Данная библиотека позволяет мне создавать оконные приложения и диалоговые окна).

Visual basic(6) неплохая альтернатива но так как я не знаю C++(7) то от данного варианта я отказываюсь в пользу PyQt6

-Использование библиотеки os(8)(Она позволяет выполнить операции связанные с файловой системой, управлением процессами, переменными окружения и многим другим)

1.3 Функциональность:

-Интерфейс(для учителя и ученика)

-Возможность ученика:

1. Прохождение теста
2. Возможность досрочно закончить тест

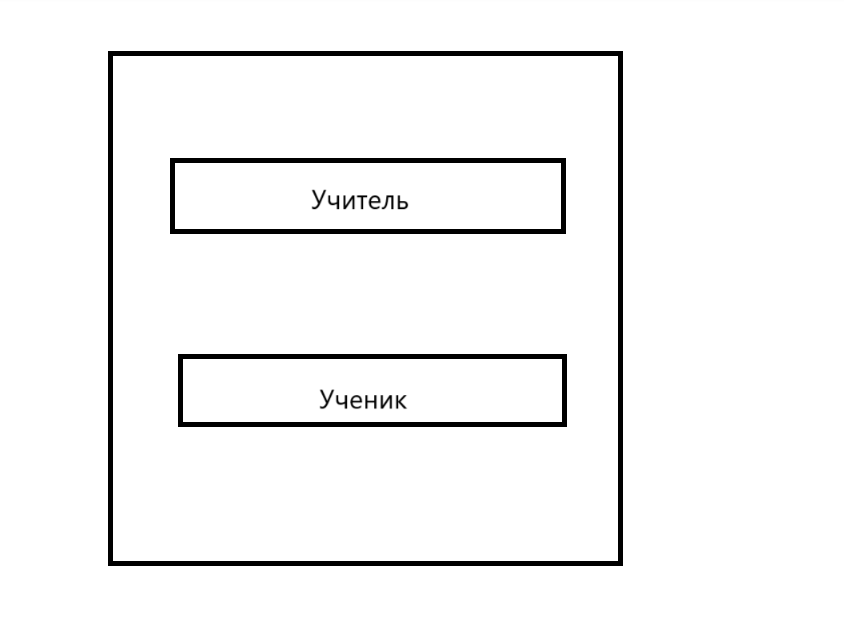
-Возможность учителя:

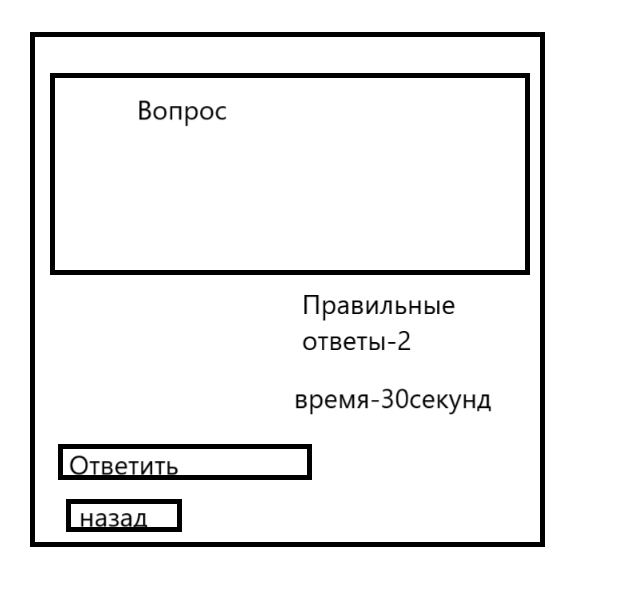
1. Создание теста
2. Добавление вопросов
3. Удаление вопросов
4. Загрузка тестов
5. Сохранение тестов
6. Проверка учеников
7. Возможность смены пароля
8. Очищение базы данных за одно нажатие

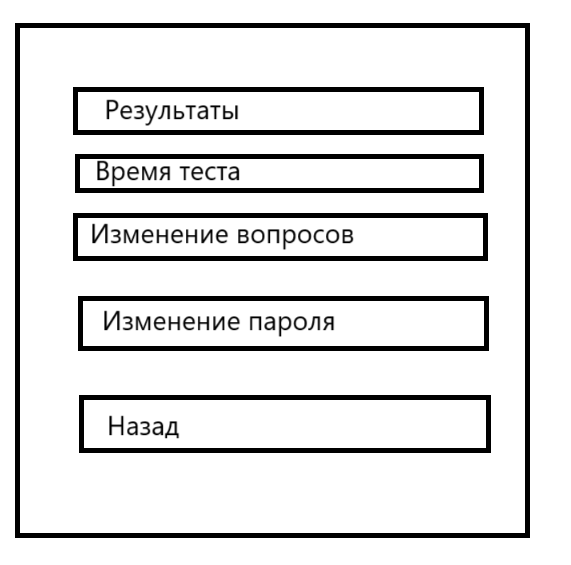
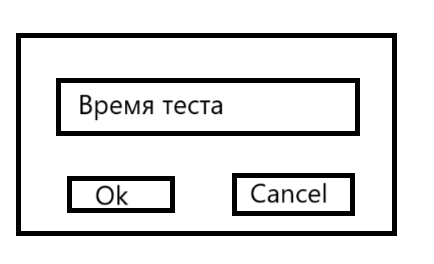
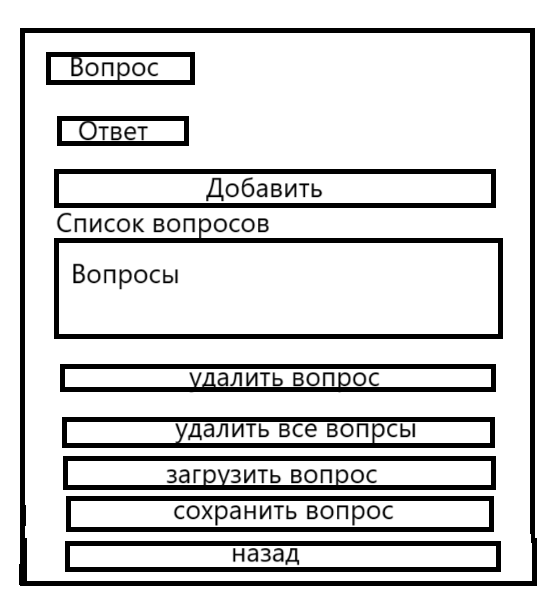
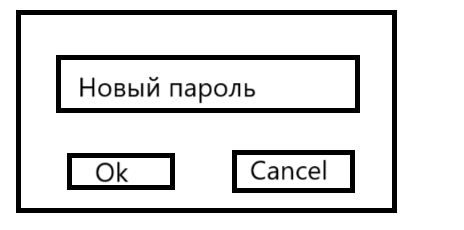
Глава 2 - Разработка проекта

2.1 Интерфейс:

1)Стартовое окно

 2)Окно ученика



1. Окно администратора 
2. Окно Результатов 
3. Message box для изменения времени 
4. Окно для изменения вопросов 
5. Message box для изменения пароля 

2.2 Базы данных:

-Так как мне необходимо сохранять вопросы, ответы, очки ученика и пароль то я буду использовать базу данных SQL-lite(9)

-Также я буду сохранять свои тесты в txt формате

Структура базы данных

Таблица questions:

id (INTEGER, PRIMARY KEY) – Уникальный идентификатор вопроса.

question\_text (TEXT) – Текст вопроса.

correct\_answer (TEXT) – Правильный ответ на вопрос.

Таблица results:

id (INTEGER, PRIMARY KEY) – Уникальный идентификатор результата.

student\_name (TEXT) – Имя студента, который проходил викторину.

score (INTEGER) – Количество правильных ответов.

created\_at (DATETIME) – Дата и время проведения викторины. Дата и время проведения викторины.

2.3 Код программы:

Основные классы

-Класс Database:

Отвечает за взаимодействие с базой данных.

Содержит методы для создания таблиц, добавления преподавателей, получения вопросов и вставки результатов.

Работает с SQLite для сохранения данных на диске.

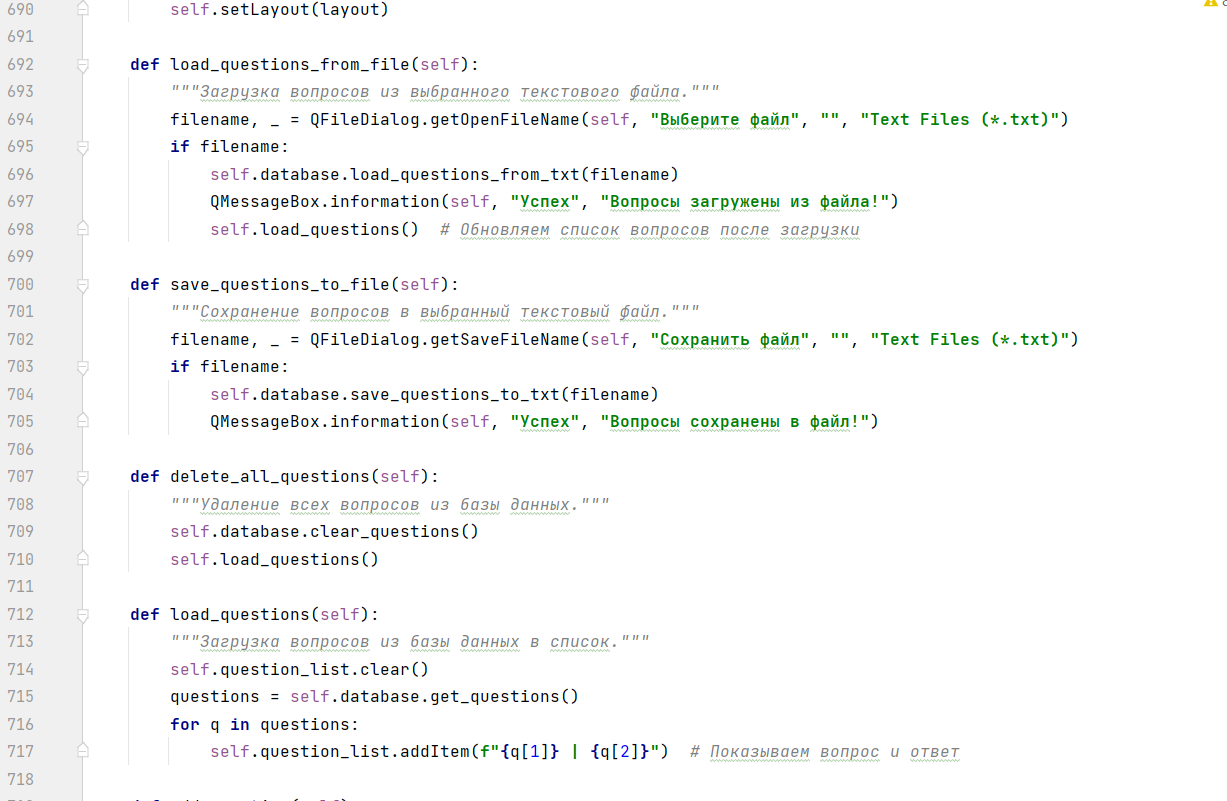
-Класс StudentWindow

(или другой класс для интерфейса студента):

Представляет окно для студента, отвечает за отображение вопросов викторины и сбор ответов.

Использует методы из класса Database для получения вопросов и записи результатов.

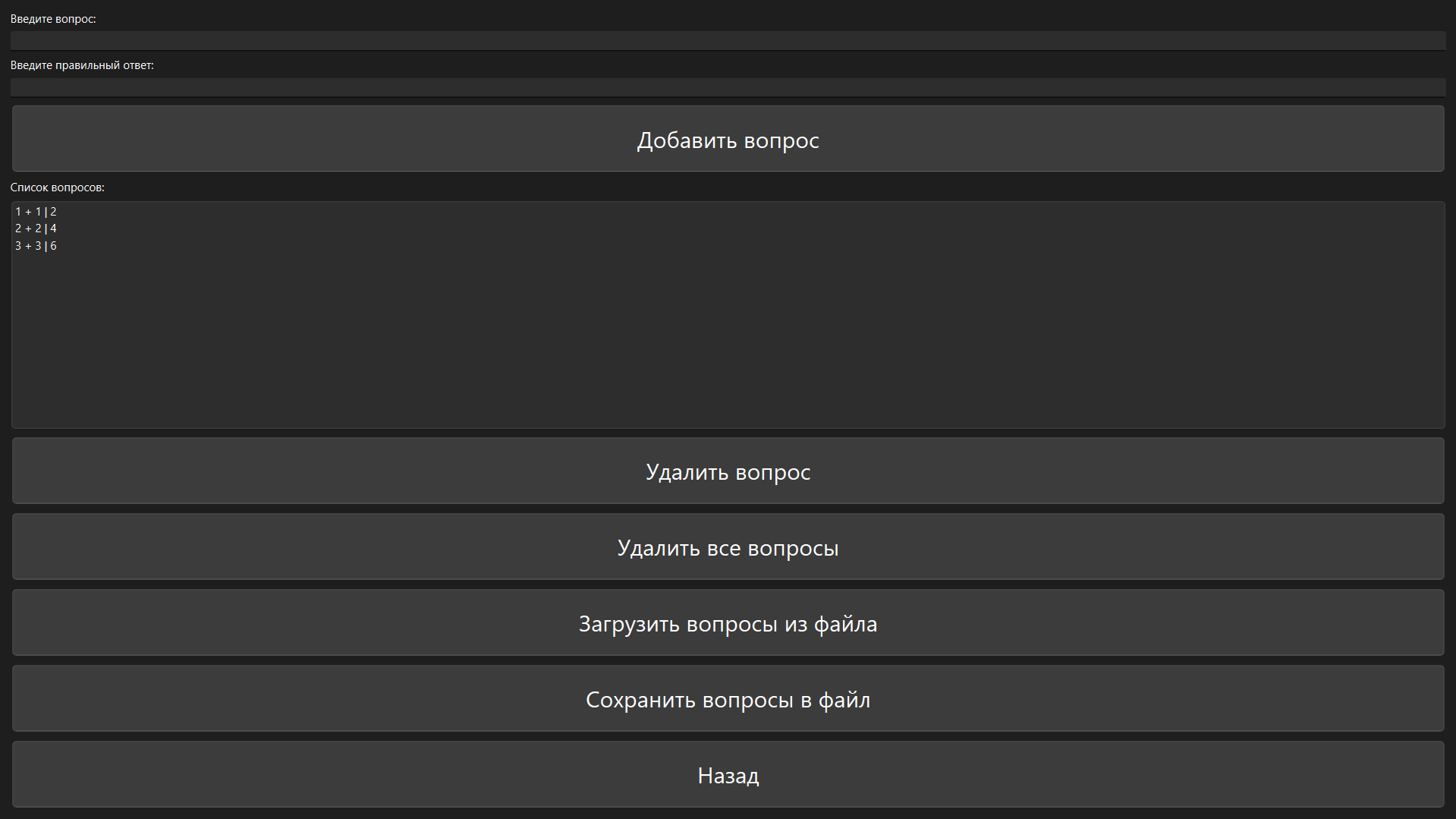
-Класс TeacherWindow (или Teacher\_win) отвечает за интерфейс и функциональность, доступные учителю. В контексте вашего приложения он может включать возможность добавления и управления вопросами, а также обновления паролей и других административных задач.



2.4 Сборка и тестирование:

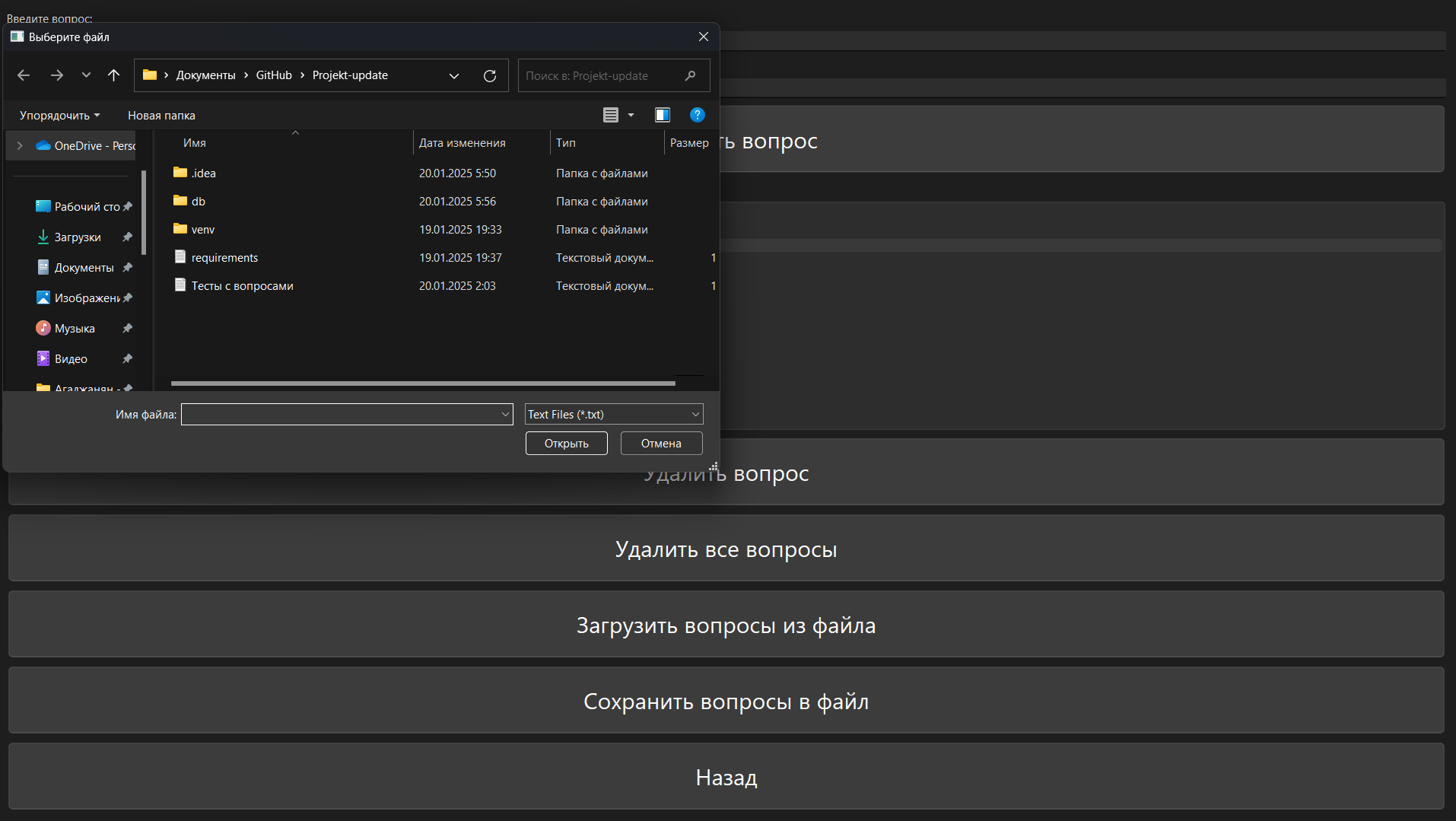
На этой стадии разработки я начал тестировать программу чтобы выявить в ней баги

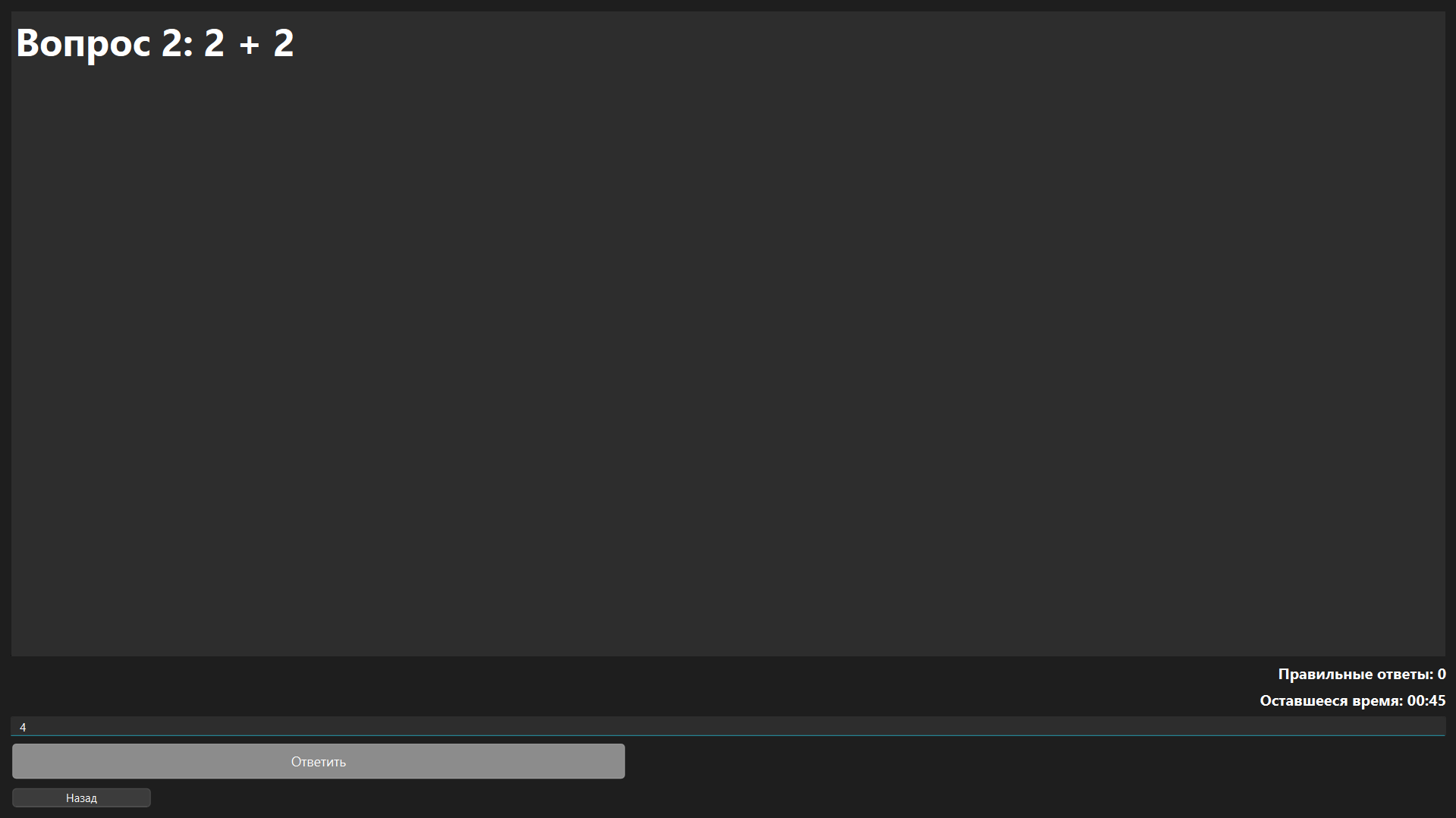
-Для начала я создал вопросы

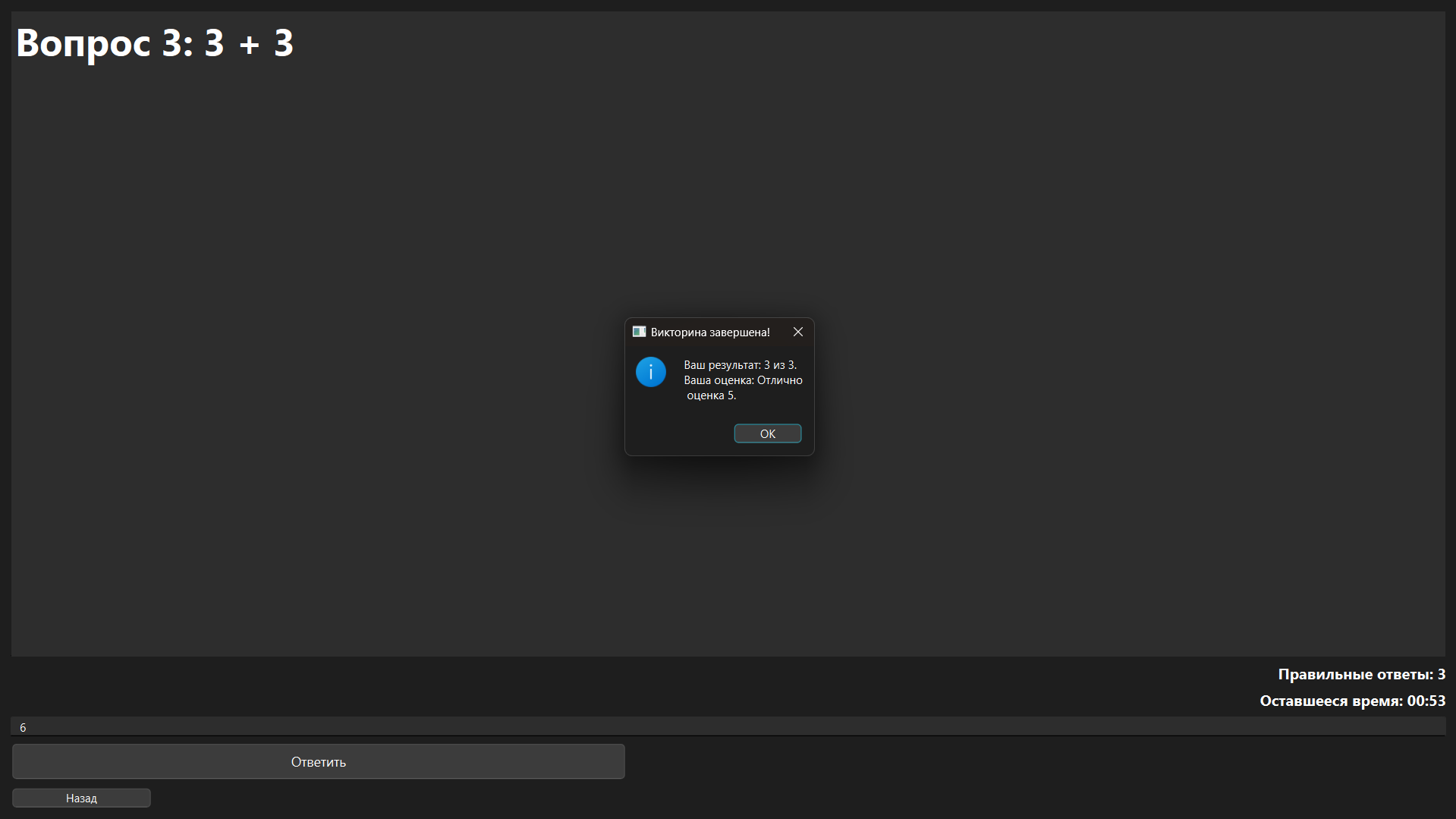


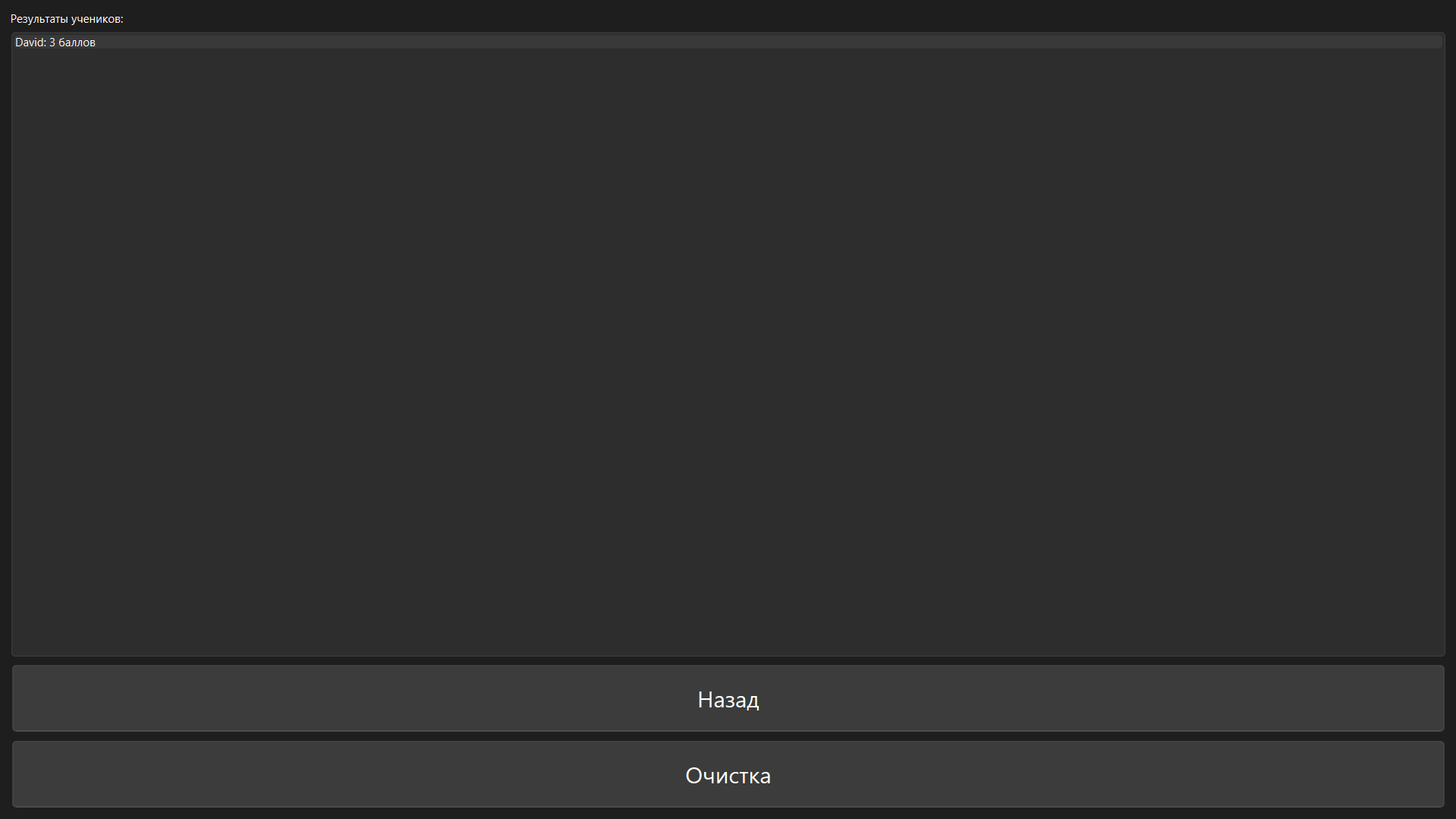
-После чего пробовал их удалять и заново добавлять

-Также я пробовал загружать вопросы



-Теперь можно решить готовый тест и получить оценку



-Также мы можем проверить свой результат в окне учителя

Отлично программа работает!!!

3 Вывод:

В процессе работы с этим проектом я изучил

Такие технологии как Python, PyQt6, Os, SQL-lite

В итоге у меня получился проект с:

-неплохим интерфейсом

-лучшей доступностью

-хорошей функциональностью

-отличной работой с базами данных

Список литературы:

1 - [Компьютерное тестирование знаний: Сетевая программа NetTest: Скачать: сайт Константина Полякова](https://kpolyakov.spb.ru/prog/nettest.htm)

2 - [MyTestXPro Wiki Help](https://mytestx.pro/wiki/%D0%97%D0%B0%D0%B3%D0%BB%D0%B0%D0%B2%D0%BD%D0%B0%D1%8F_%D1%81%D1%82%D1%80%D0%B0%D0%BD%D0%B8%D1%86%D0%B0)

3-[Online Test Pad - Онлайн тесты, опросы, кроссворды. Онлайн конструктор тестов, опросов, кроссвордов. Виджеты для вашего сайта. | Online Test Pad](https://onlinetestpad.com/)

4-[Welcome to Python.org](https://www.python.org/)

5-[Qt for Python](https://doc.qt.io/qtforpython-6/)

6-[Visual Basic docs - get started, tutorials, reference. | Microsoft Learn](https://learn.microsoft.com/en-us/dotnet/visual-basic/)

7-[C++ — Википедия](https://ru.wikipedia.org/wiki/C++)

8-[os — Miscellaneous operating system interfaces — Python 3.13.1 documentation](https://docs.python.org/3/library/os.html)

9-[SQLite — Википедия](https://ru.wikipedia.org/wiki/SQLite)

Приложения:

Можно добавить что в процессе написания я учитывал отзывы программ моих конкурентов

И вдохновлялся некоторыми их идеями и механиками для улучшения своего проекта

