государственное бюджетное профессиональное образовательное учреждение

**«Пермский химико-технологический техникум»**

Специальность 09.02.07 «Информационные системы и программирование»

**КУРСОВОЙ ПРОЕКТ**

**Междисциплинарный курс:** МДК.01.01 «Разработка программных модулей»»

**Тема:** «Разработка системы для прохождения тестирования»

Выполнил студент гр. ИС-20-11

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ (И.А. Большаков)

Руководитель проекта:

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ (Е.Г. Рыкалова)

2022

**СОДЕРЖАНИЕ**

[ВВЕДЕНИЕ 3](#_Toc116959784)

[1. Общая часть 4](#_Toc116959785)

[1.1. Описание предметной области 4](#_Toc116959786)

[1.2. Требования к функциональным характеристикам 4](#_Toc116959787)

[1.3. Требования к надежности 6](#_Toc116959788)

[1.4. Требования к техническим характеристикам 7](#_Toc116959789)

[1.5. Требования к программным характеристикам 7](#_Toc116959790)

[2. Описательная часть 8](#_Toc116959791)

[2.1. Выбор технологий проектирования. Выбор СУБД 8](#_Toc116959792)

[2.2. Построение модели системы 10](#_Toc116959793)

[2.2.1. Построение концептуальной модели 10](#_Toc116959794)

[2.2.2. Описание логической структуры 11](#_Toc116959795)

[2.2.3. Проектирование физической структуры базы данных 12](#_Toc116959796)

[2.3. Алгоритм функционирования информационной системы 13](#_Toc116959797)

[2.4. Описание пользовательского интерфейса 14](#_Toc116959798)

[2.5. Тестирование и отладка 17](#_Toc116959799)

[ЗАКЛЮЧЕНИЕ 21](#_Toc116959800)

[СПИСОК ЛИТЕРАТУРЫ 22](#_Toc116959801)

ПРИЛОЖЕНИЯ

[ТЕКСТ ПРОГРАММЫ 23](#_Toc116959802)

[РУКОВОДСТВО ОПЕРАТОРА 46](#_Toc116959803)

# **ВВЕДЕНИЕ**

Тестирование широко применяется для оценки уровня знаний в учебных заведениях. Испытуемому предлагается ряд вопросов, на которые он должен ответить. Обычно к каждому вопросу дается несколько вариантов ответа, из которых надо выбрать правильный. Каждому варианту ответа соответствует некоторая оценка. Суммированием оценок за ответы получается общий балл, на основе которого делается вывод об уровне подготовленности испытуемого.

В наше время технологии развиваются очень быстро. Но в школах все еще многие учителя сталкиваются с проблемой проведения тестирования. Учитель использует свое свободное время для создания теста, распечатки и проверки тестов учеников лично. Так же ученикам приходится ждать своей оценки. Данная проблема актуальна в наше время и будет рассмотрена в курсовом проекте.

Цель курсового проекта заключается в разработке приложения для системы тестирования, которое облегчит работу учителям.

Для достижения поставленной цели нужно решить следующие задачи:

1. Изучить предметную область;
2. Разработать базу данных по данной области;
3. Разработать начальные страницы, предназначенные для пользователей;
4. Разработать систему авторизации – регистрации, а также обеспечить разделение ролей;
5. Разработать функционал для создания новых и изменения старых тестов и просмотр пройденных тестов учениками.

## **Общая часть**

### **Описание предметной области**

Приложение установлено на компьютерах в компьютерных кассах школы или же на личные компьютеры учеников. Тестирование представляет собой интерактивную систему для самопроверки знаний учащегося. Контрольные вопросы с вариантами выбора ответов. Программа может контролировать ответы учащихся.

Администратор должен зарегистрировать учителей и учеников в данном приложении. В базе данных администратор хранит фамилии пользователей, их логины и пароли, и email.

Учитель должен разработать тест с ответами для учеников. В истории прохождении тестов учитель видит кто прошел тест, какой тест был выполнен, когда он был завершен и сколько баллов было получено.

Тест в свое время состоит из нескольких вопросов и вариантов ответов на вопросы.

Ученик должен пройти тестирование и после прохождения он может увидеть сообщении об успешном прохождении теста. После прохождения теста на его email будет отправлены результаты пройденого теста.

* 1. **Требования к функциональным характеристикам**

База данных должна осуществлять следующие функции:

- хранение и защита данных;

Приложение по работе с базой данных должна осуществлять следующие функции:

- осуществлять добавление, изменение и удаление данных, хранящихся в базе данных;

- просмотр существующих данных о тестах, вопросы, ответов на вопросы и пользователях;

Данная программа должна облегчить работу учителям при создании тестов и проверки знаний учеников. При авторизации/регистрации пользователь вводит свои данные и происходит вход.

На выходе мы должны получить уведомление об успешном прохождении тестирования и отправку письма на почту.

Основные функции информационной системы представлены на рис. 1.

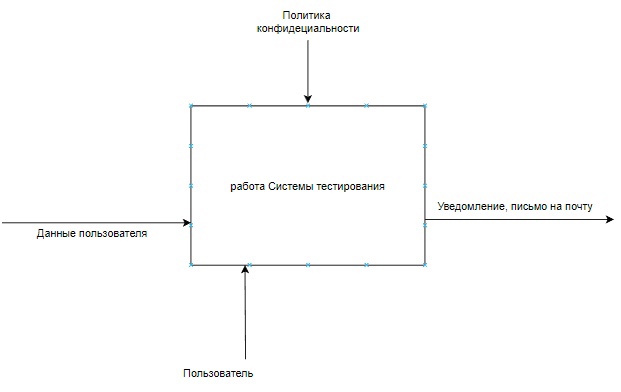


Рис. 1

Для пользователей основные функции продемонстрированы на рис. 2.

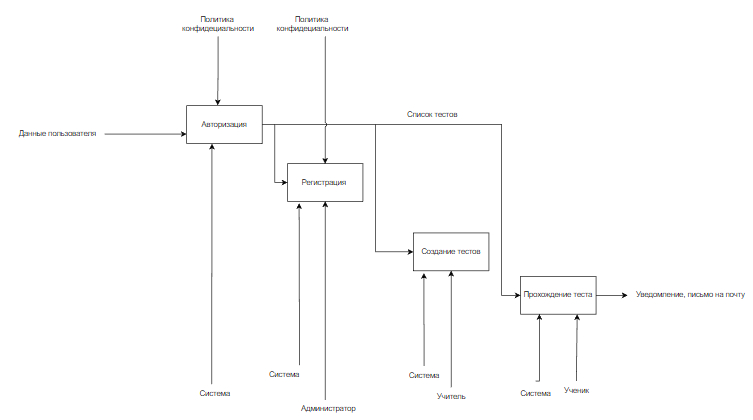
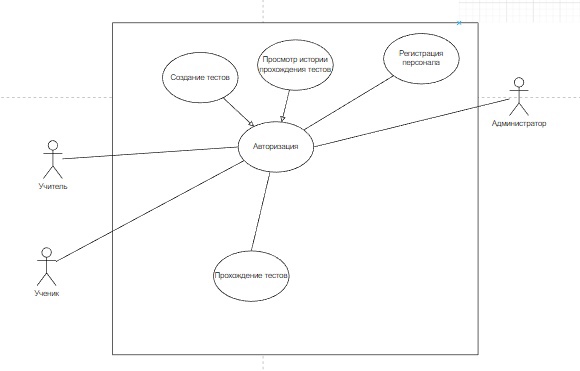


Рис. 2

* 1. **Требования к надежности**

В приложении присутствует разграничение ролей: ученик, учителя и администратор. У каждого пользователя свой уникальный логин и пароль, которые доступны только ему. Данная разработка защищает личные данные пользователя.

Доступные функции к системе и подсистеме для отдельных ролей отображены на рис. 3.

  
Рис. 3

* 1. **Требования к техническим характеристикам**

Для выполнения программы необходим персональный компьютер. Который состоит из системного блока, монитора, клавиатуры, мыши или ноутбук. Так же потребуется принтер для печати билета.

Компьютер или ноутбук должен имеет следующие минимальные технические характеристик:

* 1 ГБ свободного места на жестком диске;
* разрешение экрана не меньше 1024х768 точек
* не менее 512 мб оперативной памяти
  1. **Требования к программным характеристикам**

Данная программа будет разработана для операционных систем выше Window’s 7.

1. **Описательная часть**
   1. **Выбор технологий проектирования. Выбор СУБД**

Проектирование баз данных – процесс создания схемы базы данных и определения необходимых ограничений целостности.

Основные задачи базы данных:

* Обеспечение хранения в БД всей необходимой информации;
* Обеспечение возможности получения данных по всем необходимым запросам;
* Сокращение избыточности и дублирования данных;
* Обеспечение целостности базы данных.

Семантическое проектирование — это подход к реализации приложения, целью которого является создание логической иерархии базы данных, формирование структуры приложения.

Концептуальное проектирование — начальный этап проектирования, при котором создается черновой вариант продукта, показывающий принцип закладываемой логики для первоначального согласования и проверки ограничений средств разработки. Концептуальный прототип создается без направления на конкретную СУБД и модель данных.

Логическое проектирование – это процесс создания модели используемой на предприятии информации на основе выбранной модели организации данных, но без учета типа целевой СУБД и других физических аспектов реализации. Для реляционной модели данных логическая модель — набор схем отношений, обычно с указанием первичных ключей, а также «связей» между отношениями, представляющих собой внешние ключи.

Физическое проектирование — создание схемы базы данных для конкретной СУБД. Специфика конкретной СУБД может включать в себя ограничения на именование объектов базы данных, ограничения на поддерживаемые типы данных и другие параметры.

Выполнив все вышеперечисленные этапы поможет в полной мере выполнить все задачи и разработать базу данных, которая удовлетворит всем обязательным требованиям для ее администрирования и внедрения в работу.

Для разработки базы данных нужно выбрать конкретную СУБД.

Можно выделить ряд критериев, по которым следует проводить выбор СУБД:

* модель данных;
* дополнительные возможности;
* особенности архитектуры и функциональные возможности;
* особенности разработки приложений;
* производительность;
* надежность;

Существуют множество различных баз данных, между которыми нужно выбрать одну. База данных должна удовлетворять все критерия и быть удобной. Рассмотрим 3 основные базы данных:

SQLite — это быстрая и легкая встраиваемая одно файловая СУБД на языке C, которая не имеет сервера и позволяет хранить всю базу локально на одном устройстве. Для работы SQLite не нужны сторонние библиотеки или службы.

Microsoft SQL Server — система управления реляционными базами данных, разработанная корпорацией Microsoft. Основной используемый язык запросов — Transact-SQL. Используется для работы с базами данных размером от персональных до крупных баз данных масштаба предприятия.

PostgreSQL - это свободно распространяемая объектно-реляционная система управления базами данных (ORDBMS), наиболее развитая из открытых СУБД в мире и являющаяся реальной альтернативой коммерческим базам данных.

Преимущества SQLite заключаются в файловой системе, то есть все данные хранятся в одном файле. SQLite использует язык запросов SQL, но не все функции доступны. Так же в этой СУБД есть свои недостатки, такие как отсутствие возможности управлять связями в таблицах и невозможность дополнительной настройки.

PostgreSQL имеет полную SQL-совместимость, расширяемость, объектно-ориентированность. Но у PostgreSQL есть проблемы с производительностью и данное СУБД не так распространено, как, например, SQL servrer.

Microsoft SQL Server самая популярная СУБД и достаточна проста в использовании. Имеет много функций, хорошую безопасность, производительность.

Сравнив все СУБД по критериям, можно понять, что Microsoft SQL Server больше походит для решения задач курсового проекта в силу своей доступности и удобства в использовании.

* 1. **Построение модели системы**
     1. **Построение концептуальной модели**

Концептуальная модель представляет собой высокоуровневый взгляд на предметную область. На данном этапе не учитывается модель данных и физические аспекты представления и хранения данных, проектирование одинаково для любой базы данных. ER-диаграмма показана на рис. 4.

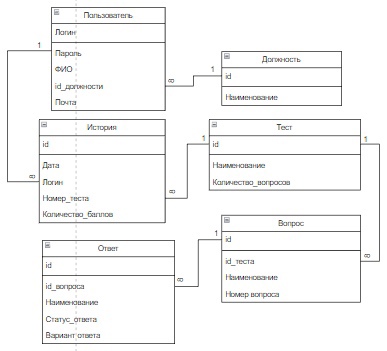


Рис. 4

* + 1. **Описание логической структуры**

Логическая модель описывает понятия предметной области, их взаимосвязь, а также ограничения на данные, налагаемые предметной областью. Графическое представление логической структуры представлено на рис. 5.

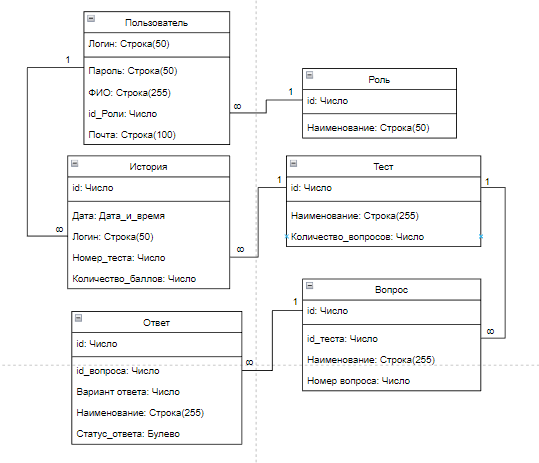


Рис. 5

* + 1. **Проектирование физической структуры базы данных**

Физическая модель базы данных декларирует и описывает возможность размещения и связи данных в текущей среде хранения, а также способы доступа. Физическая модель представлена на рис. 6.

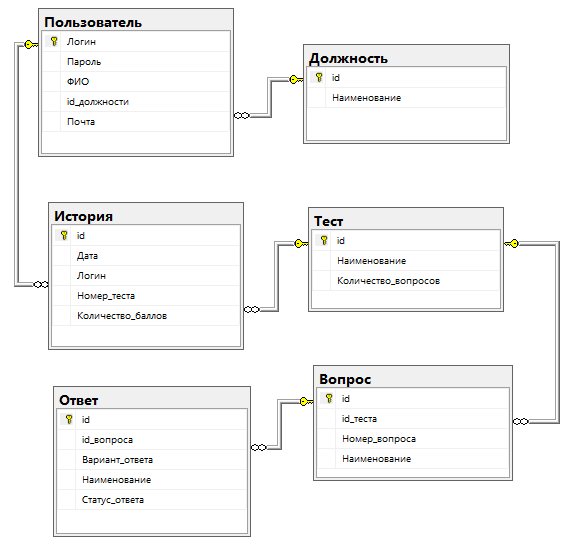


Рис. 6

* 1. **Алгоритм функционирования информационной системы**

Алгоритм функционирования информационной системы представлен в виде блок-схемы, которая представлена на рис. 7.

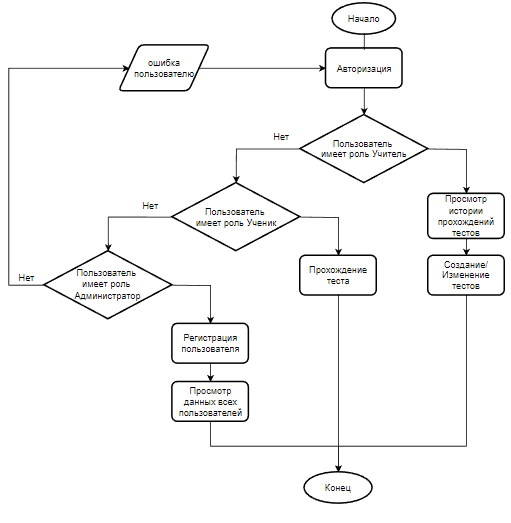


Рис. 7

* 1. **Описание пользовательского интерфейса**

Пользовательский интерфейс программы – это связующее звено между пользователем и компьютером, выполняющим эту программу.

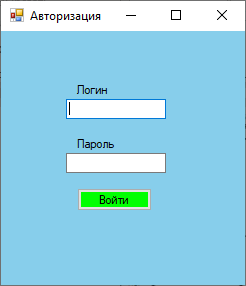


Рис. 8 Окно авторизации

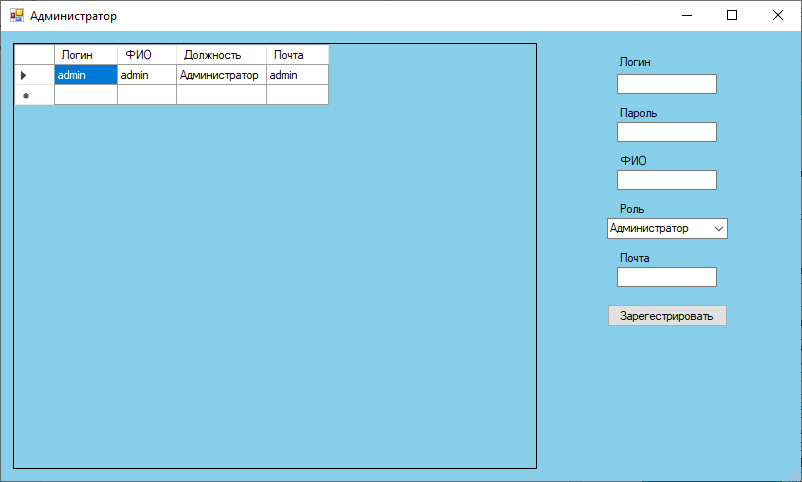


Рис.9 Панель администратора

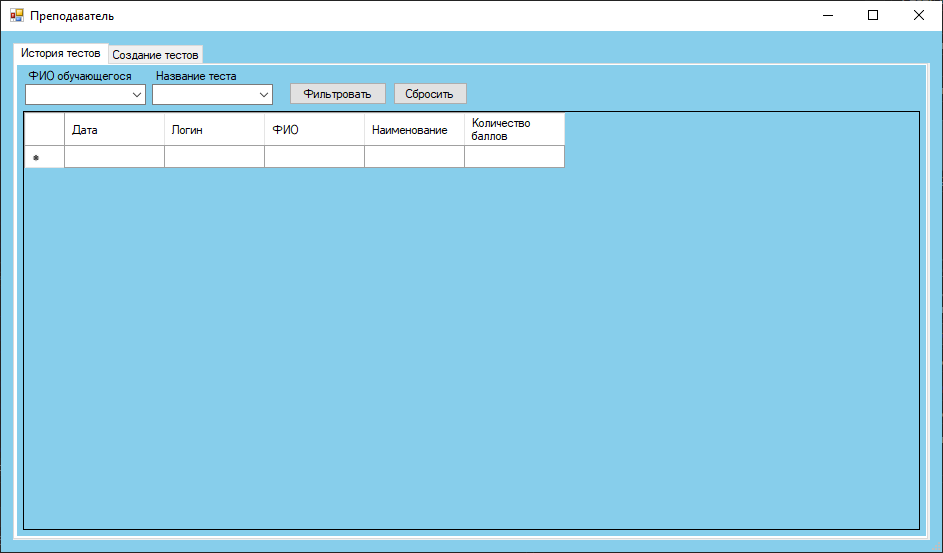


Рис. 10 Панель учителя для просмотра истории тестов

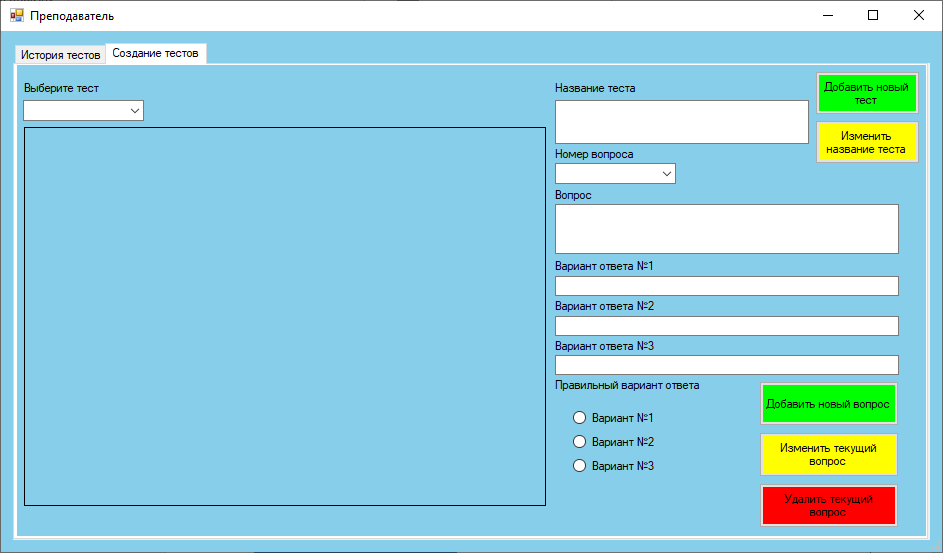


Рис. 11 Панель учителя для создания тестов

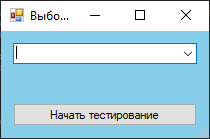


Рис. 12 Панель выбор теста для прохождения

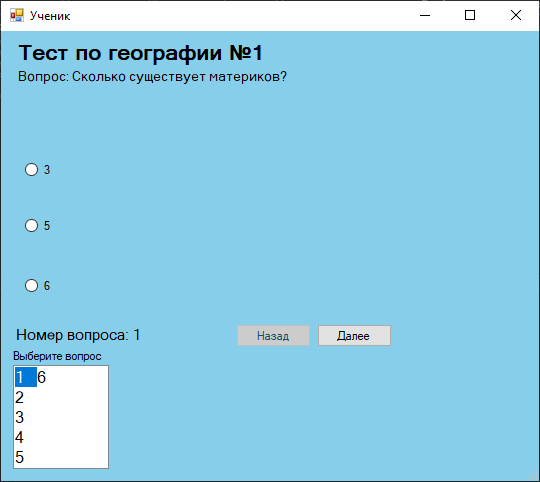


Рис. 13 Панель прохождения теста

* 1. **Тестирование и отладка**

Данное приложение было протестирована в нормальных, экстремальных и исключительных условиях.

Тестирование в нормальных условиях:

|  |  |
| --- | --- |
| **Тест Case #** | 1 |
| **Приоритет тестирования** | Высокий |
| **Заголовок/название теста** | Тестирование процедуры регистрации нового Пользователя |
| **Краткое изложение теста** | Ввод корректных данных в текстовые поля на форме «Регистрация нового сотрудника» |
| **Этапы теста** | 1. Запуск программы  2. Авторизоваться под ролью Администратора  3. Заполнить поля данными  4. Нажать на кнопку «Зарегистрировать» |
| **Тестовые данные** | Логин – «igor»  Пароль – «igor»  ФИО – «Большаков Игорь Алексеевич»  Роль– «Ученик»  Email – «volnas.111@gmail.com» |
| **Ожидаемый результат** | Вывод сообщения об успешной регистрации |
| **Фактический результат** | Рис. 14 |
| **Статус** | Pass |
| **Предварительное условие** | Запуск программы, нажать на кнопку «Регистрация» |
| **Постусловие** | Вывод сообщения об успешной регистрации сотрудника. |
| **Примечания/комментарии** | На почту зарегистрированного пользователя отправляется письмо на почту с его учетными данными |

Тестирование в экстремальных условиях:

|  |  |
| --- | --- |
| **Тест Case #** | 2 |
| **Приоритет тестирования** | Высокий |
| **Заголовок/название теста** | Тестирование процедуры незаполненных полей в текстовых полях на форме «Авторизациии» |
| **Краткое изложение теста** | Незаполненные поля логина и пароля в окне «авторизации» |
| **Этапы теста** | 1. Запуск программы  2. Нажатие на кнопку «Вход» |
| **Тестовые данные** | Логин – «»  Пароль – «» |
| **Ожидаемый результат** | Вывод сообщения о неправильно введенных данных. |
| **Фактический результат** | Рис. 15 |
| **Статус** | Pass |
| **Предварительное условие** | Запуск программы, авторизация |
| **Постусловие** | Вывод сообщения об ошибке. |
| **Примечания/комментарии** |  |

Тестирование в исключительных условиях:

|  |  |
| --- | --- |
| **Тест Case #** | 3 |
| **Приоритет тестирования** | Высокий |
| **Заголовок/название теста** | Авторизация в системе неизвестного пользователя |
| **Краткое изложение теста** | Ввод неправильных данных в поля ввода |
| **Этапы теста** | 1. Запуск программы  2. Ввод логина и пароля пользователя в текстовое поле  3. Нажатие на кнопку «Вход» |
| **Тестовые данные** | Логин – test  Пароль – test |
| **Ожидаемый результат** | Вывод сообщения о неправильно введенных данных. |
| **Фактический результат** | Рис. 16 |
| **Статус** | Pass |
| **Предварительное условие** | Запуск программы, авторизация |
| **Постусловие** | Вывод сообщения об ошибке. |

В нормальных условиях, тестирование ошибок не выявило. В исключительных и экстремальных условиях тестирование также не выявило проблем, при несоблюдении каких-либо условий, система выводит ошибку о несоблюдении данного условия.

# **ЗАКЛЮЧЕНИЕ**

Разработанное приложение позволяет учителям создавать тест и просматривать историю прохождения теста. Администратор создавать новых пользователей в системе. А ученик может проходить тест и просматривать свои результаты на почте.

Так же решены все ранее поставленные задачи:

* проанализирована предметная область;
* спроектирована и реализована базы данных для хранения информации;
* разработана авторизация для разграничения прав доступа;
* разработано приложение для работы с базой данных.

Для правильного функционирования информационной системы для разработки были использованы такие системы как, интегрированная среда разработки Visual Studio и системы управления реляционными базами данных MS SQL Server.

Таким образом, система обеспечивает возможность добавления, изменения и удаления данных в базе данных и имеет удобный интерфейс для работы пользователей. Программный продукт удобен и легок в использовании благодаря разработанному интерфейсу.

# **СПИСОК ЛИТЕРАТУРЫ**

1. Мюллер Д. П. C# для чайников: учебное пособие – Москва: Диалектика 2019 г. – 609 с.
2. Tproger — всё о программировании [электронный ресурс] / адрес: https://tproger.ru/translations/sqlite-mysql-postgresql-comparison/ Дата обращения: 05.10.2022
3. Metanit. Полное руководство по C# 10 и .NET 6 [электронный ресурс] / адрес: https://metanit.com/sharp/tutorial/ Дата обращения: 08.10.2022
4. Metanit. Руководство по MS SQL Server 2019 [электронный ресурс] / адрес: https://metanit.com/sql/sqlserver/1.1.php/ Дата обращения: 08.10.2022

**ПРИЛОЖЕНИЕ 1**

## **ТЕКСТ ПРОГРАММЫ**

**Код класса «что-то»:**

**ПРИЛОЖЕНИЕ 2**

**РУКОВОДСТВО ОПЕРАТОРА**

1. **Назначение программы**
   1. **Функциональное назначение программы**

Главная задача приложения — это разработка приложения для системы тестирования, которое облегчит работу учителям.

* 1. **Эксплуатационное назначение программы**

Данное приложение будет эксплуатировано учениками, учителями, а также администратором. Приложение и база данных предназначена, как и для использования администраторами, так и для домашнего использования.

* 1. **Состав функций**

Приложение выполняет такие функции как:

* Получение необходимой информации из базы данных;
* Редактирование, удаление и добавление данных, хранимых в базе данных, посредством интерфейса приложения;
* Авторизация в системе и выдача прав.
* Рассылка писем на почту.

1. **Условия выполнения программы**
   1. **Минимальный состав аппаратных средств**

Для работы с программным продуктом компьютер должен быть на ОС Windows 8/8,1/10/11, иметь минимальные требования и доступ к интернету.

* 1. **Минимальный состав программных средств**

Данный программный продукт разрабатывается под платформу Windows. Программа должна работать под основными версиями этой платформы: Windows8, Windows8.1, Windows10, Windows11. Так же дополнительно потребуется Microsoft SQL Server.

* 1. **Требования к персоналу**

Для эксплуатации данного приложения требуются умения и навыки работы с компьютером на уровне пользователя.

1. **Выполнение программы**
   1. **Загрузка и запуск программы**

После запуска приложения, пользователь попадает на главное окно с «Авторизацией». Для авторизации, пользователю необходимо ввести данные в поля «Логин» и «Пароль» после чего нажать кнопку «Войти».

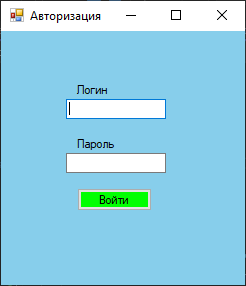


Рис. 17

* 1. **Вход в приложение с правами доступа «Администратор»**
     1. После авторизации, пользователь попадает на главное окно «Администратора» (Рис.18).

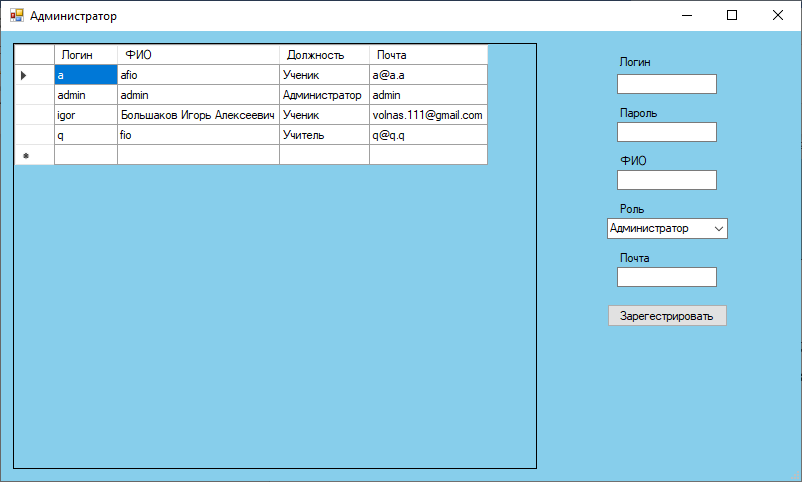
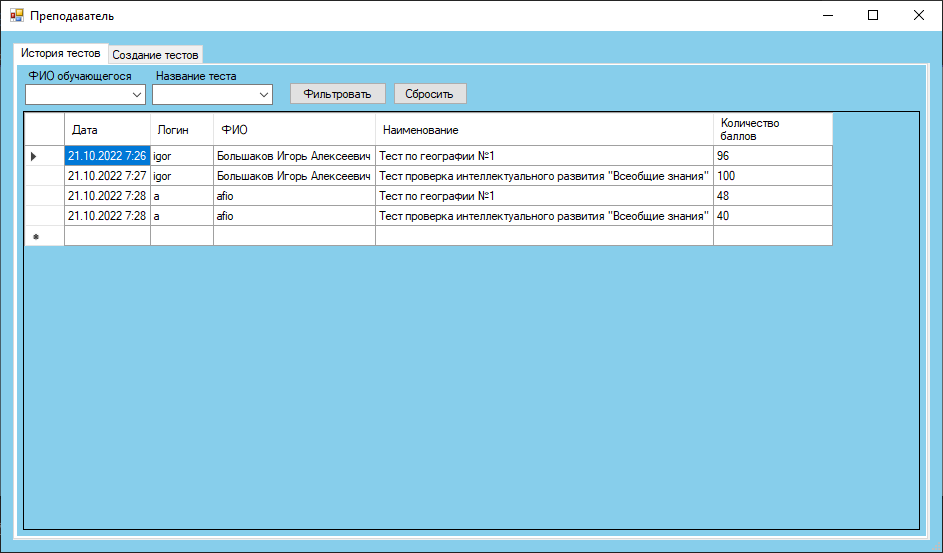
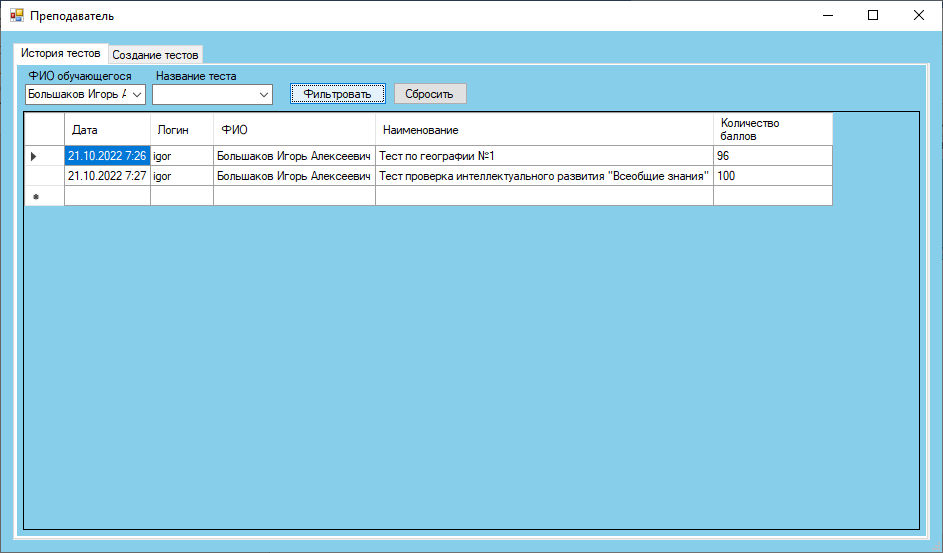
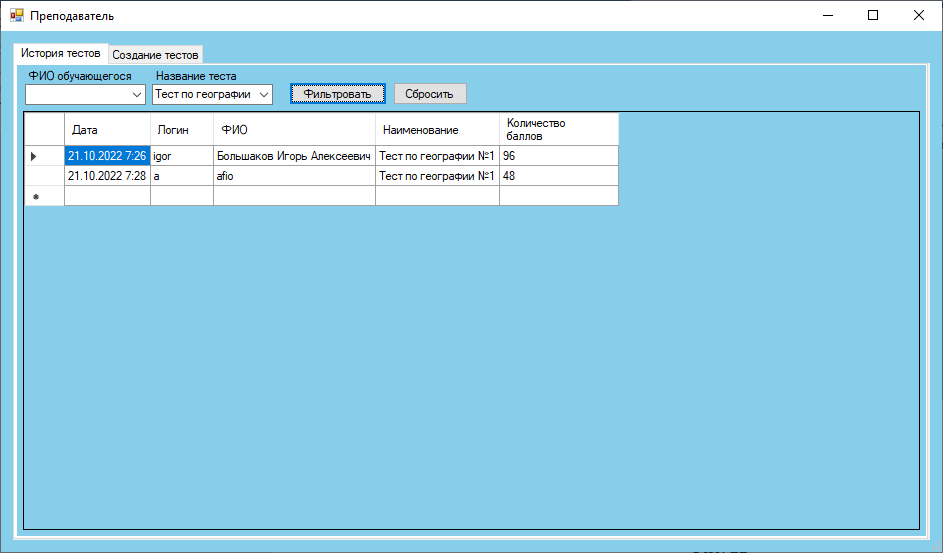
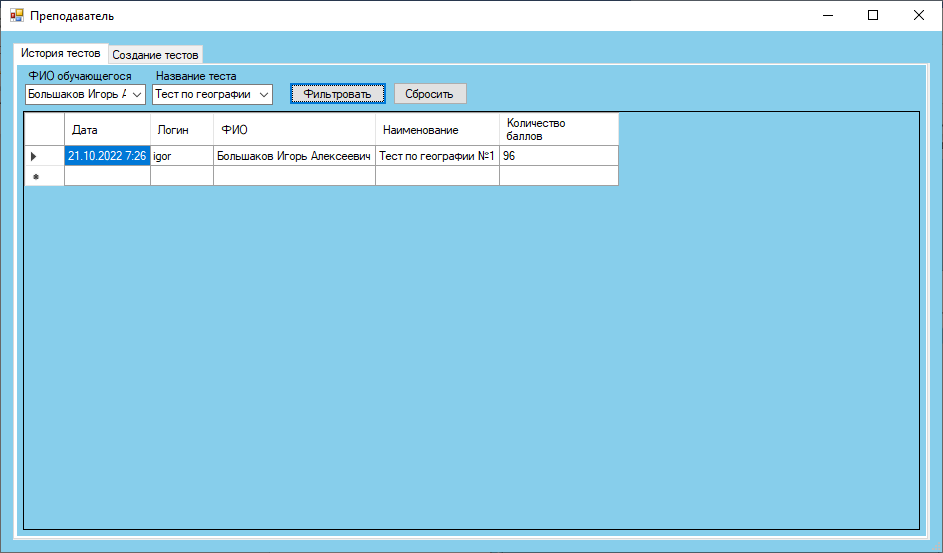


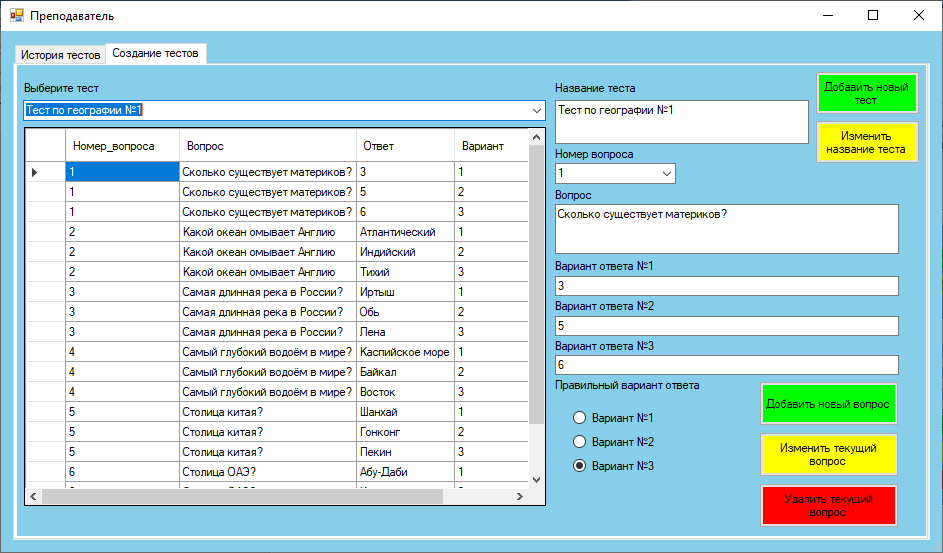
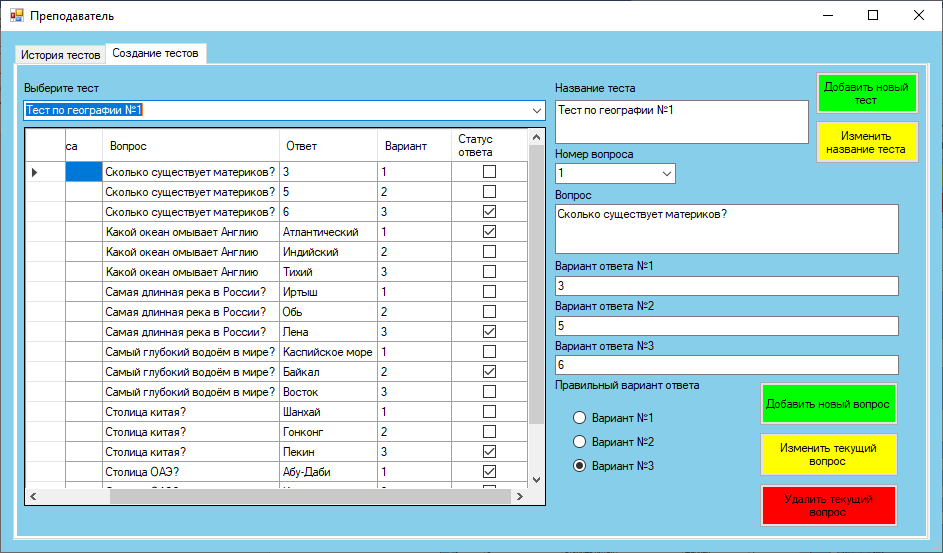
Рис. 18

* + 1. Пользователь может зарегистрировать нового пользователя в базу данных заполнив все поля, выбрав роль и нажать на кнопку «Зарегистрировать» (Рис.18).
  1. **Вход в приложение с правами доступа «Учитель»**
     1. После авторизации, пользователь попадает на главное окно «Учитель» (Рис.19). На вкладке «История тестов», пользователь видит историю всех пройденных тестов (Рис.19). Пользователь может отфильтровать список по «Наименовании теста» и «ФИО пользователя» при помощи выбора нужных данных в выпадающих списках и нажатии на кнопку «Фильтровать» (Рис. 20, Рис. 21, Рис. 22), также пользователь может сбросить фильтрацию путём нажатия кнопки «Сбросить» (Рис.19).

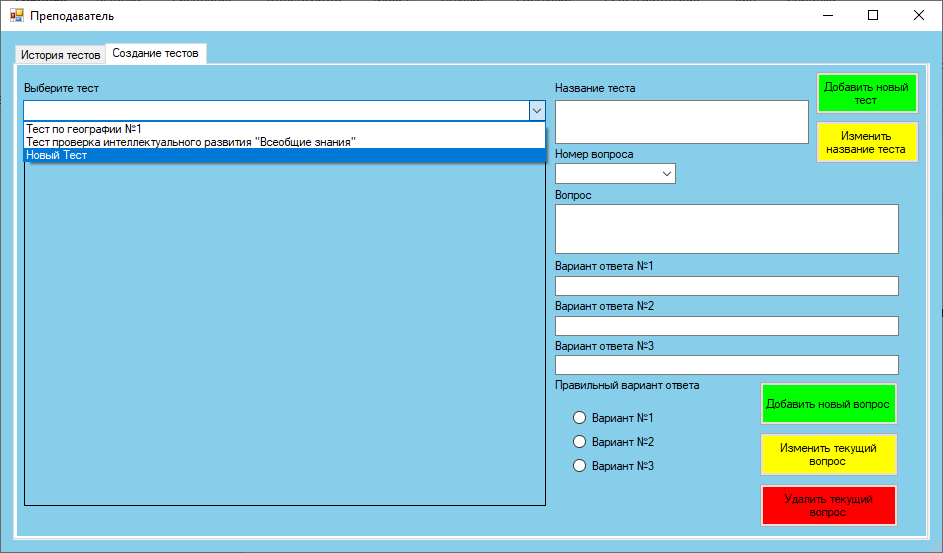
Рис. 19

  
Рис. 20  
  
Рис. 21  
  
Рис. 22

* + 1. На вкладке «Создание теста» пользователь видит интерфейс по созданию тестов (рис. 23). При выборе существующего теста, в таблицу выводятся ответы и вопросы на этот тест (Рис. 24),

  
Рис. 23  
  
Рис. 24

* + 1. Для создания нового теста, пользователю необходимо заполнить поле «Название теста» после чего нажать на зеленую кнопку «Добавить новый тест», после чего выбрать новый созданный тест в выпадающем списке (Рис. 25, Рис. 26).

  
Рис. 25

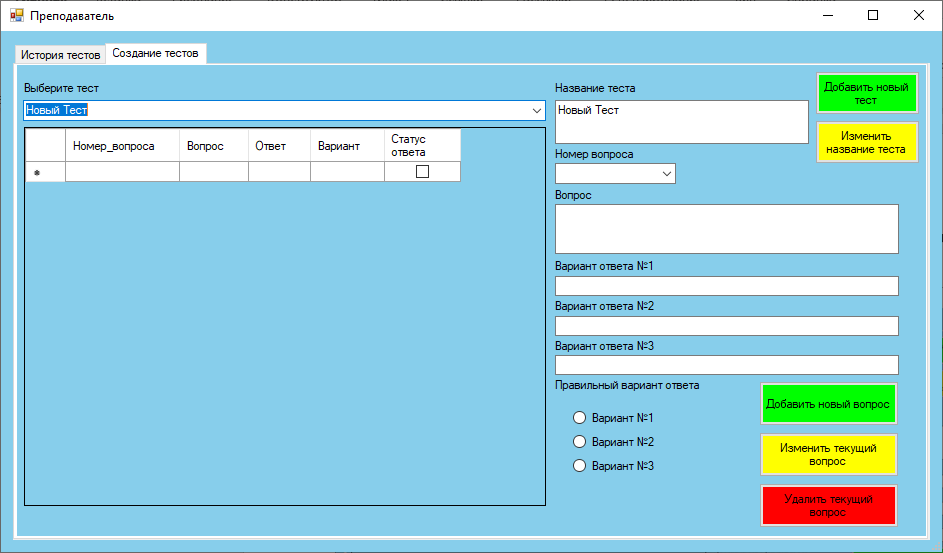


Рис. 26

* + 1. Для изменения названия теста пользователю необходимо выбрать тест, изменить поле «Название теста» и нажать на жёлтую кнопку «Изменить название теста» после чего выбрать его из выпадающего списка с тестами для просмотра информации о нём (Рис. 27).

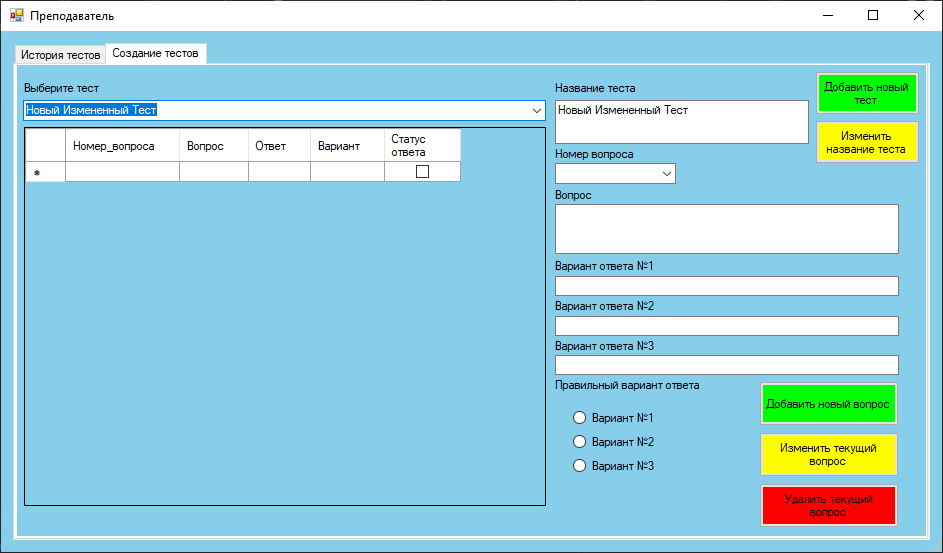
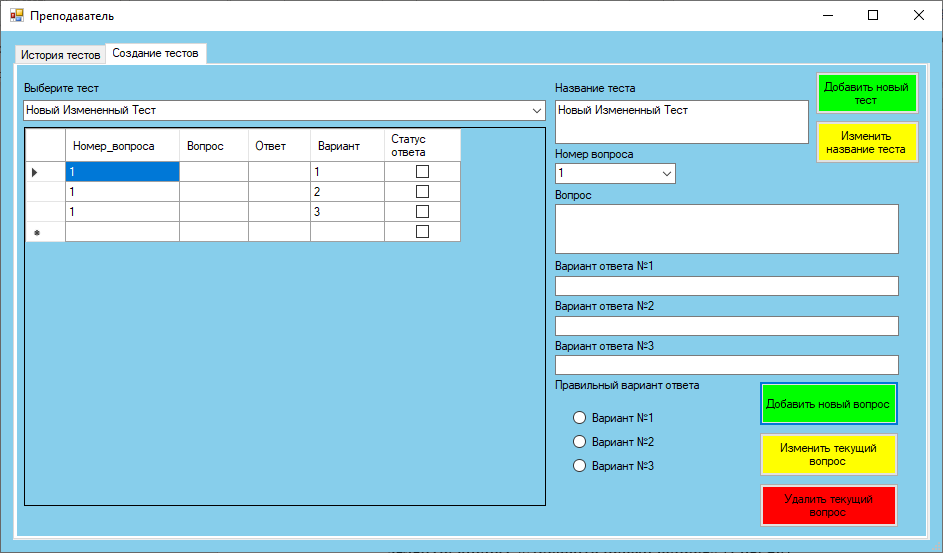
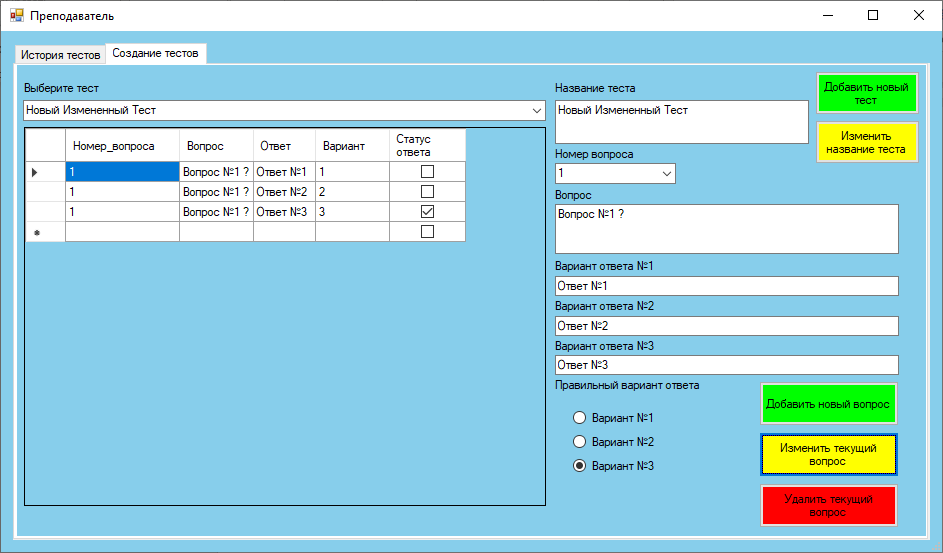


Рис. 27

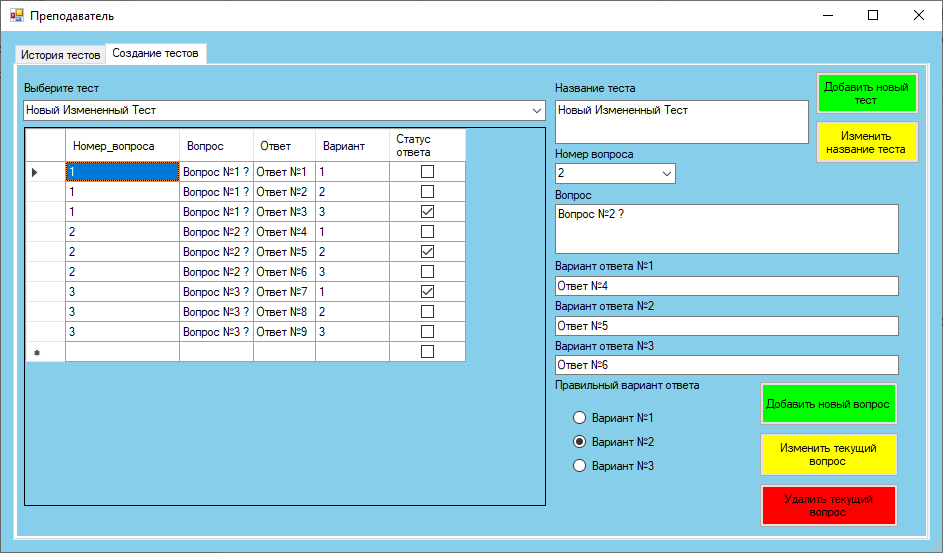
* + 1. Для добавление нового вопроса, пользователю нужно нажать на зелёную кнопку «Добавить новый вопрос» (Рис. 28)

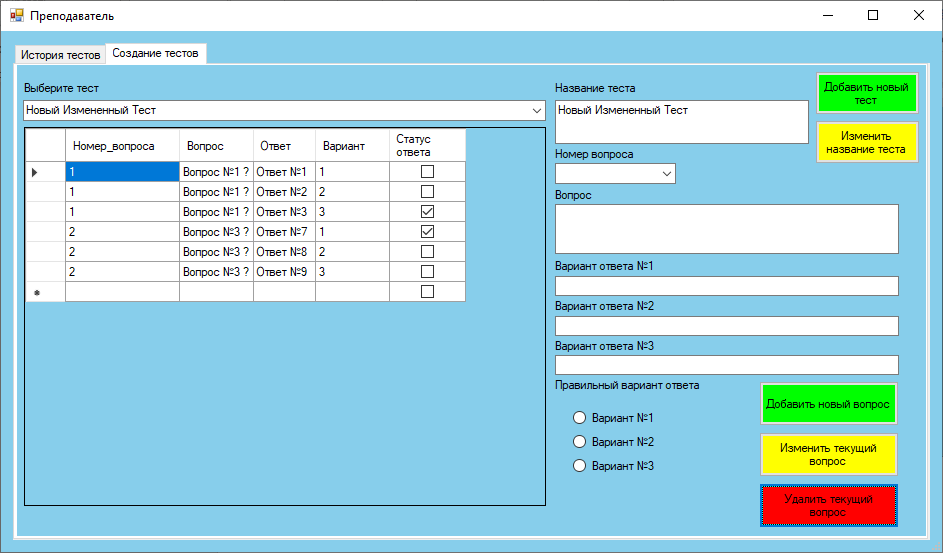
  
Рис. 28

* + 1. После добавления вопроса пользователем, следует его изменить и не оставлять пустым, для этого пользователю необходимо будет выбрать номер вопроса в выпадающем списке «Номер вопроса», заполнить все поля, выбрать правильный вариант ответа и нажать на жёлтую кнопку «Изменить текущий вопрос» (Рис. 29). Также можно изменять уже существующее вопросы, изменив соответствующие поля. При выборе вопроса данные в поля заносятся автономно.

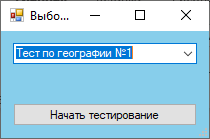
  
Рис. 29

* + 1. Пользователь может удалить вопрос из теста путём выбора теста в выпадающем списке и нажатием на красную кнопку «Удалить текущий вопрос» (Рис. 30, Рис. 31).

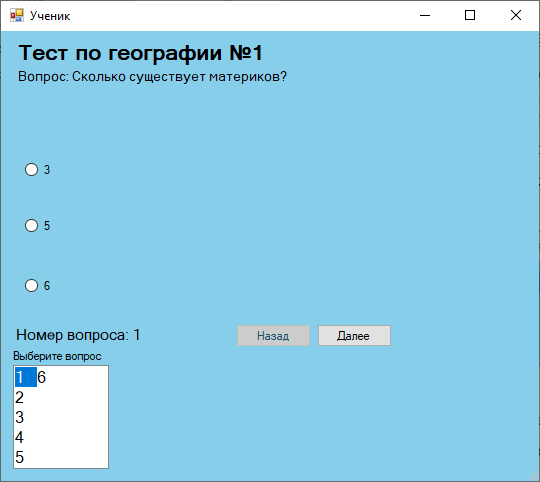
  
Рис. 30

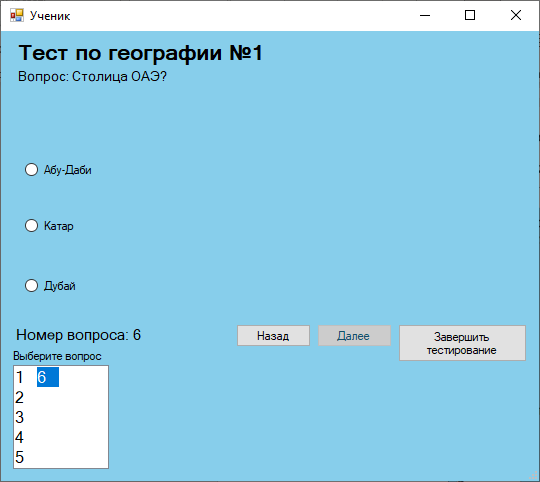
  
Рис. 31

* 1. **Вход в приложение с правами доступа «Ученик»**
     1. После авторизации, пользователь попадает на окно с выбором тестов (Рис. 32). Для прохождения тестирования пользователю нужно выбрать в выпадающем списке название теста и нажать на кнопку «Начать тестирование»

  
Рис. 32

* + 1. После выбора теста пользователь попадает на главное окно по прохождению тестов (Рис. 33). В этом окне пользователю необходимо отвечать на заданные вопросы путём нажатия на переключатели с вариантами ответов. Навигация по вопросам осуществляется путем нажатием кнопок «Далее» или «Назад», а также щелчком мыши на номер вопроса находящийся в списке в левом нижнем углу приложения. По достижении последнего вопроса появляется кнопка «Завершить тестирование» (Рис. 24), после нажатия которой тестирование будет остановлено, а результат будет отправлен на почту проходившему тест пользователю и записан в «Историю прохождения тестирования».

  
Рис. 33

  
Рис. 34

* 1. **Завершение работы программы**

Чтобы завершить работу программы необходимо закрыть приложение путём нажатия на крест в правом верхнем углу приложения.

1. **Сообщения оператору**
   1. **Сообщения**

Сообщение пользователю при создании нового теста (Рис. 35).

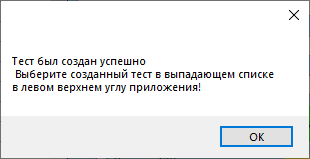


Рис. 35

Сообщение пользователю при попытке зарегистрировать пользователя с уже существующим логином в базе данных (Рис. 36).

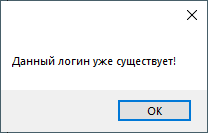


Рис. 36

Сообщение об окончании тестирования (Рис. 37).

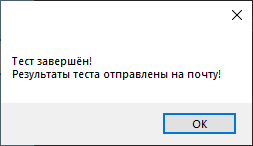


Рис. 37

Письмо на почту новому пользователю с данными для входа (Рис. 38).

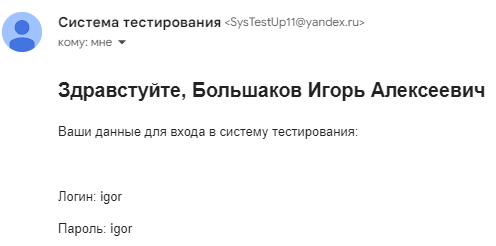


Рис. 38

Письмо на почту пользователя, завершившего тестирование, с результатами пройденного тестирования (Рис. 39).

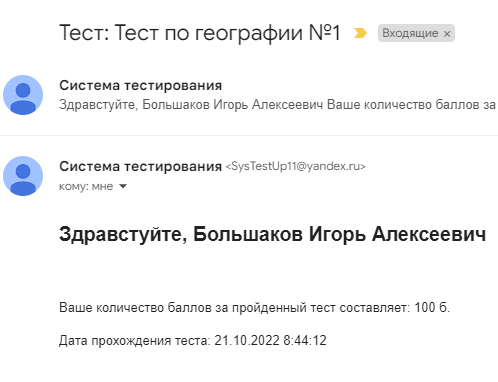


Рис. 39

Сообщение пользователю при попытке войти с несуществующего аккаунта (Рис. 40).

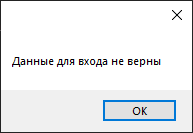


Рис. 40

Сообщение пользователю при неверной почте во время регистрации   
(Рис. 41).

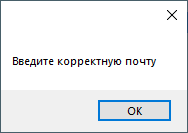


Рис. 41