**СОДЕРЖАНИЕ**

[ВВЕДЕНИЕ 2](#_Toc140530801)

[ГЛАВА 1 ПРОЕКТИРОВАНИЕ ИНФОРМАЦИОННОЙ СИСТЕМЫ ОЦЕНКА ОСТАТОЧНЫЙ ЗНАНИЙ ОБУЧАЮЩИХСЯ 3](#_Toc140530802)

[1.1. Разработка эскизного проекта 3](#_Toc140530803)

[1.2. Программное обеспечение для разработки информационной системы 7](#_Toc140530804)

[ГЛАВА 2 РАЗРАБОТКА ИНФОРМАЦИОННОЙ ОЦЕНКА ОСТАТОЧНЫЙ ЗНАНИЙ ОБУЧАЮЩИХСЯ 11](#_Toc140530805)

[2.1 Разработка базы данных 11](#_Toc140530806)

[2.2 Разработка программного обеспечения 15](#_Toc140530807)

[ЗАКЛЮЧЕНИЕ 28](#_Toc140530808)

# **ВВЕДЕНИЕ**

В настоящее время СПК СКГА динамично развивается, внедряя в образовательный процесс современные образовательные технологии и инновационные практики воспитательной работы. Миссия колледжа – подготовка компетентных, конкурентно-способных выпускников, готовых к непрерывному профессионально-личностному саморазвитию, ориентированных на профессиональный успех и непрерывное образование в быстро меняющемся мире.

Актуальностью выбранной темы является развитие компьютерных и информационных технологии, которые могут поспособствовать развитию колледжа как инновационной площадке необходимо развивать информационную структуру в учебном заведении.

Целью разрабатываемой информационной система является автоматизировать бизнес-процессы СПК СКГА, что позволит увеличить не только эффективность, но и качество при работе с информацией.

Для достижения поставленных требований были поставлены следующие задачи:

1. Ознакомиться с структурой организации СПК СКГА;
2. Разработать эскизный проект;
3. Составить техническое задание;
4. Построить базу данных;
5. Разработать пользовательский интерфейс;

Предметом исследования является информационной система - система, предназначенная для хранения, поиска и обработки информации, и соответствующие организационные ресурсы, которые обеспечивают и распространяют информацию.

Объектом исследования является колледж СПК СКГА. Необходимо изучить структуру предприятия, собрать все необходимые данные для составления и заполнения базы данных.

**ГЛАВА 1 ПРОЕКТИРОВАНИЕ ИНФОРМАЦИОННОЙ СИСТЕМЫ ОЦЕНКА ОСТАТОЧНЫЙ ЗНАНИЙ ОБУЧАЮЩИХСЯ**

* 1. **Разработка эскизного проекта**

Диаграмма не принадлежит к семантическим элементам языка UML и её значение не зависит от того, каким образом она представлена. Кроме редких исключений основная часть несомой ею информации содержится не в размере или расположении её элементов, а в её топологической конфигурации. Чаще всего применяются три основных типа визуальных отношений между элементами: соединение (в виде линий от одной двумерной формы к другой), включение (в виде помещения одних форм внутрь других) и визуальное подкрепление (в виде размещения одних символов недалеко от других). Возможно также представить данную нотацию и в аналитической форме как совокупность связей узлов графа.

Язык моделирования UML предоставляет своему пользователю большое количество предопределённых разновидностей диаграмм. Как правило, тип каждой диаграммы определяется большинством элементов, которые она отображает. Однако, ничто не мешает проектировщику определить и свой собственный вид диаграммы исходя из требований данной конкретной задачи.

В большинстве ситуаций, для представления статических частей модели используются структурные диаграммы, а для предоставления её динамической части применяются поведенческие диаграммы [4].

Диаграмма вариантов использования ([англ.](https://ru.wikipedia.org/wiki/%D0%90%D0%BD%D0%B3%D0%BB%D0%B8%D0%B9%D1%81%D0%BA%D0%B8%D0%B9_%D1%8F%D0%B7%D1%8B%D0%BA) use case diagram) в [UML](https://ru.wikipedia.org/wiki/UML) — диаграмма, отражающая отношения между [актёрами](https://ru.wikipedia.org/wiki/%D0%90%D0%BA%D1%82%D0%BE%D1%80_(UML)) и [прецедентами](https://ru.wikipedia.org/wiki/%D0%9F%D1%80%D0%B5%D1%86%D0%B5%D0%B4%D0%B5%D0%BD%D1%82_(UML)) и являющаяся составной частью модели прецедентов, позволяющей описать систему на концептуальном уровне.

На данной диаграмме изображены варианты трех видов пользователей ПО: тестер, студент, администратор.

тестер имеет доступ к форме редактирования и способен добавлять изменять и удалять тесты.

Пользователь имеет доступ к форме прохождения и не способен редактировать информацию из базы данных.

Администратор редактирует информацию в базе данных, и имеет доступ ко всем таблицам.

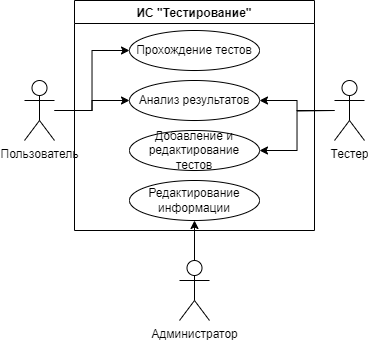


Рисунок 1 – «Диаграмма прецедентов»

Диаграмма последовательности (рис.2) *(*[*англ.*](https://ru.wikipedia.org/wiki/%D0%90%D0%BD%D0%B3%D0%BB%D0%B8%D0%B9%D1%81%D0%BA%D0%B8%D0%B9_%D1%8F%D0%B7%D1%8B%D0%BA)*sequence diagram)* — [UML-диаграмма](https://ru.wikipedia.org/wiki/%D0%94%D0%B8%D0%B0%D0%B3%D1%80%D0%B0%D0%BC%D0%BC%D0%B0_(UML)), на которой для некоторого набора объектов на единой временной оси показан жизненный цикл объекта (создание-деятельность-уничтожение некой сущности) и взаимодействие актеров (действующих лиц) информационной системы в рамках [прецедента](https://ru.wikipedia.org/wiki/%D0%9F%D1%80%D0%B5%D1%86%D0%B5%D0%B4%D0%B5%D0%BD%D1%82_(UML)).

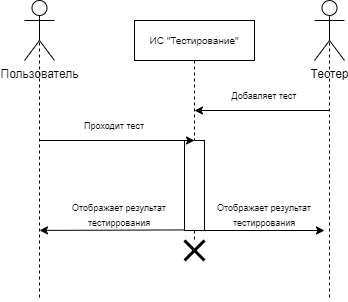


Рисунок 2 – «Диаграмма последовательности»

Диаграмма классов (рис.3) ([англ.](https://ru.wikipedia.org/wiki/%D0%90%D0%BD%D0%B3%D0%BB%D0%B8%D0%B9%D1%81%D0%BA%D0%B8%D0%B9_%D1%8F%D0%B7%D1%8B%D0%BA) *class diagram*)  — структурная  [диаграмма](https://ru.wikipedia.org/wiki/%D0%94%D0%B8%D0%B0%D0%B3%D1%80%D0%B0%D0%BC%D0%BC%D0%B0_(UML))  языка моделирования  [UML](https://ru.wikipedia.org/wiki/UML), демонстрирующая общую структуру иерархии [классов](https://ru.wikipedia.org/wiki/%D0%9A%D0%BB%D0%B0%D1%81%D1%81_(%D0%BF%D1%80%D0%BE%D0%B3%D1%80%D0%B0%D0%BC%D0%BC%D0%B8%D1%80%D0%BE%D0%B2%D0%B0%D0%BD%D0%B8%D0%B5)) системы, их коопераций, [атрибутов](https://ru.wikipedia.org/wiki/%D0%9F%D0%BE%D0%BB%D0%B5_%D0%BA%D0%BB%D0%B0%D1%81%D1%81%D0%B0) (полей), [методов](https://ru.wikipedia.org/wiki/%D0%9C%D0%B5%D1%82%D0%BE%D0%B4_(%D1%8F%D0%B7%D1%8B%D0%BA%D0%B8_%D0%BF%D1%80%D0%BE%D0%B3%D1%80%D0%B0%D0%BC%D0%BC%D0%B8%D1%80%D0%BE%D0%B2%D0%B0%D0%BD%D0%B8%D1%8F)), интерфейсов и взаимосвязей между ними. Широко применяется не только для документирования и визуализации, но также для конструирования посредством прямого или обратного проектирования.

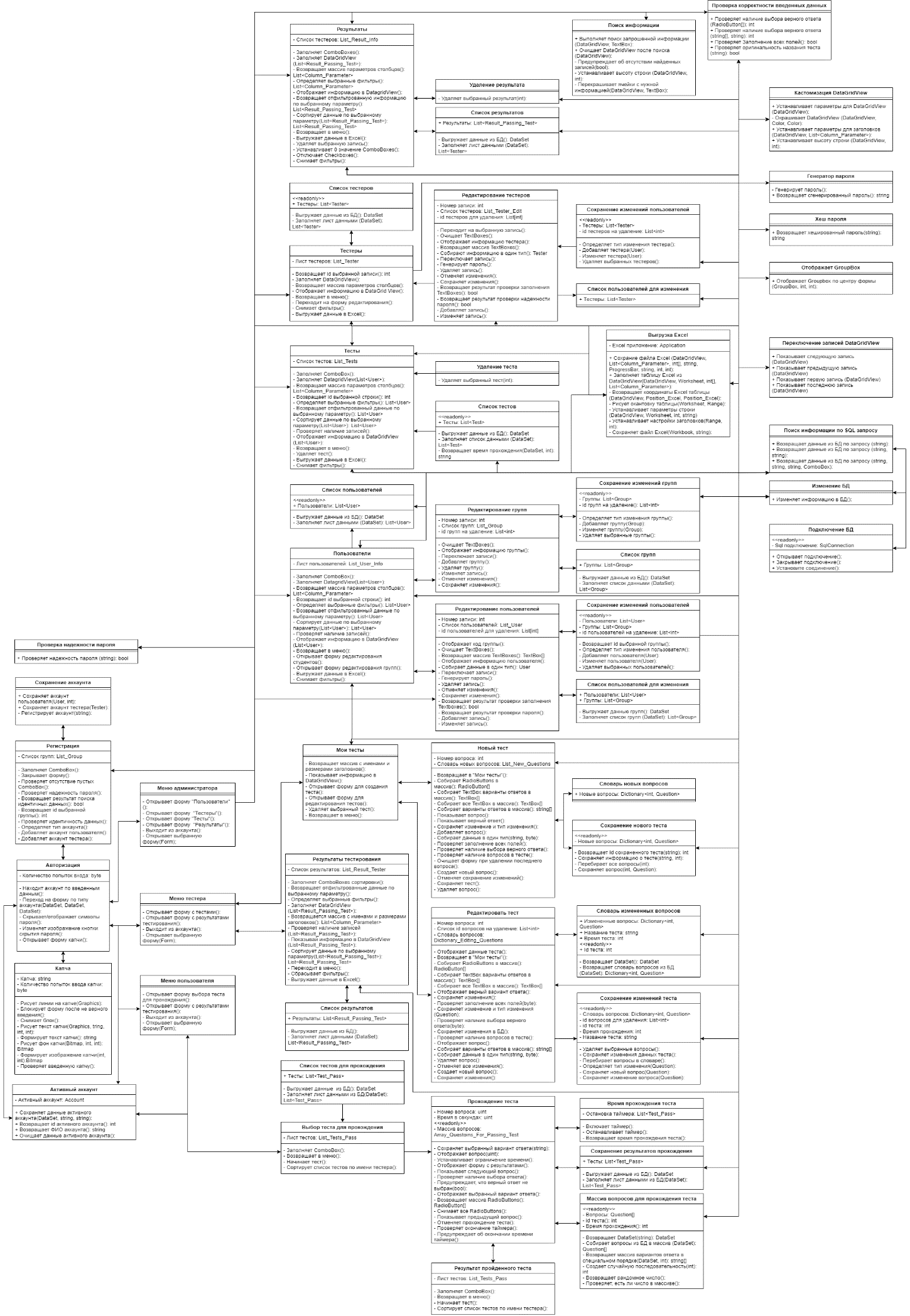


Рисунок 3 – «Диаграмма классов»

* 1. **Программное обеспечение для разработки информационной системы**

Средства разработки программного обеспечения – совокупность приемов, методов, методик, а также набор инструментальных программ (компиляторы, прикладные/системные библиотеки и т.д.), используемых разработчиком для создания программного кода Программы, отвечающего заданным требованиям. Разработка программ – сложный процесс, основной целью которого является создание, сопровождение программного кода, обеспечивающего необходимый уровень надежности и качества. Для достижения основной цели разработки программ используются средства разработки программного обеспечения.

В ходе работы над практикой, для разработки приложения, были рассмотрены следующие платформы для разработки приложений:

1. SQL;

SQL — декларативный язык программирования, применяемый для создания, модификации и управления данными в реляционной базе данных, управляемой соответствующей системой управления базами данных [14].

Является, прежде всего, информационно-логическим языком, предназначенным для описания, изменения и извлечения данных, хранимых в реляционных базах данных. В общем случае SQL (без ряда современных расширений) считается языком программирования не полным по Тьюрингу, но вместе с тем стандарт языка спецификацией SQL/PSM предусматривает возможность его процедурных расширений [12].

Изначально SQL был основным способом работы пользователя с базой данных и позволял выполнять следующий набор операций:

* создание в базе данных новой таблицы;
* добавление в таблицу новых записей;
* изменение записей;
* удаление записей;
* выборка записей из одной или нескольких таблиц (в соответствии с заданным условием);
* изменение структур таблиц.

Со временем SQL усложнился — обогатился новыми конструкциями, обеспечил возможность описания и управления новыми хранимыми объектами (например, индексы, представления, триггеры и хранимые процедуры) — и стал приобретать черты, свойственные языкам программирования [9].

При всех своих изменениях SQL остаётся самым распространённым лингвистическим средством для взаимодействия прикладного программного обеспечения с базами данных. В то же время современные СУБД, а также информационные системы, использующие СУБД, предоставляют пользователю развитые средства визуального построения запросов [7].

2. C#;

C# (произносится си шарп) — объектно-ориентированный язык программирования. Разработан в 1998—2001 годах группой инженеров компании Microsoft под руководством Андерса Хейлсберга и Скотта Вильтаумота как язык разработки приложений для платформы Microsoft .NET Framework и .NET Core. Впоследствии был стандартизирован как ECMA-334 и ISO/IEC 23270 [15].

C# относится к семье языков с C-подобным синтаксисом, из них его синтаксис наиболее близок к C++ и Java. Язык имеет статическую типизацию, поддерживает полиморфизм, перегрузку операторов (в том числе операторов явного и неявного приведения типа), делегаты, атрибуты, события, переменные, свойства, обобщённые типы и методы, итераторы, анонимные функции с поддержкой замыканий, LINQ, исключения, комментарии в формате XML [16].

Переняв многое от своих предшественников — языков C++, Delphi, Модула, Smalltalk и, в особенности, Java — С#, опираясь на практику их использования, исключает некоторые модели, зарекомендовавшие себя как проблематичные при разработке программных систем, например, C# в отличие от C++ не поддерживает множественное наследование классов (между тем допускается множественная реализация интерфейсов).

3. Visual Studio 2022;

Интегрированная среда разработки Visual Studio [6] — это стартовая площадка для написания, отладки и сборки кода, а также последующей публикации приложений. Интегрированная среда разработки (IDE) представляет собой многофункциональную программу, которую можно использовать для различных аспектов разработки программного обеспечения. Помимо стандартного редактора и отладчика, которые существуют в большинстве сред IDE, Visual Studio включает в себя компиляторы, средства автозавершения кода, графические конструкторы и многие другие функции для упрощения процесса разработки [1].

Основные инструменты, которые вам, скорее всего, понадобятся:

Обозреватель решений позволяет просматривать файлы кода, перемещаться по ним и управлять ими. Обозреватель решений позволяет упорядочить код путем объединения файлов в решения и проекты.

В окне редактора, где вы, скорее всего, будете проводить большую часть времени, отображается содержимое файла. Здесь вы можете редактировать код или разрабатывать пользовательский интерфейс, например, окно с кнопками или текстовые поля [2].

В окно вывода Visual Studio отправляет уведомления, такие как сообщения об отладке и ошибках, предупреждения компилятора, сообщения о состоянии публикаций и многие другие. Каждый источник сообщений имеет собственную вкладку.

Team Explorer позволяет отслеживать рабочие элементы и использовать код совместно с другими пользователями с помощью технологий управления версиями, таких как Git и система управления версиями Team Foundation (TFVC) [9].

4. MySQL 2019;

MySQL создавалась силами шведских разработчиков из одноименной компании в 1994 году. Тогда и состоялся ее релиз под свободной лицензией. Позже компанию поглотила Oracle [10]. MySQL распространяется бесплатно и входит в стандартный набор утилит LAMP для разработки сайтов на базе Linux.

MySQL — не единственная в своем роде. Подобных программ хватает. Но системы управления базами данных частенько ассоциируют конкретно со шведской разработкой. Доходит до того, что серверы баз данных с любым ПО называют MySQL. Все благодаря ее популярности и признанности среди крупных корпораций. Ее используют в Facebook, YouTube, Google и тысячах других IT-компаний [8].

MySQL — это популярная СУБД с моделью «клиент-сервер». Поэтому часто можно услышать от других разработчиков словосочетание «MySQL-сервер». Это действительно сервер, только сервер базы данных. Он создан для обеспечения доступа к данным для других сервисов и приложений.

В нашем случае данные — это файлы, а база — место, где они хранятся. Данные могут принимать любой облик [12].

Представьте, что вы сделали плейлист в Spotify и добавили туда новый трек. Плейлист будет базой, а добавленная песня — данными. Каждая композиция, появившаяся в уже готовом плейлисте, станет частью существующей базы данных. Так она будет пополняться.

Таких баз может быть много. Например, интернет-магазин, продающий гитары, тоже оперирует базами данных. В них хранятся инструменты из каталога, отсортированные по заранее продуманным критериям (производитель, количество струн и т.п.).

Получается, что базы могут быть многоуровневыми и делиться на различные категории (разделы каталога в случае с магазином). Так формируются взаимосвязи между разными элементами базы данных, появляется структура. Отсюда и термин «реляционная» – он намекает на зависимость элементов друг от друга [13].

Так что база данных — это набор структурированных данных с выстроенными между ними «взаимоотношениями» (делением на категории, к примеру).

Визуально она представляет собой таблицу с тысячами элементов (ссылками, файлами, отрывками текста и т.п.). Чтобы этим добром управлять, необходимо как-то обозначить таблицы и научиться ими управлять. Тут и пригодится SQL.

**ГЛАВА 2 РАЗРАБОТКА ИНФОРМАЦИОННОЙ ОЦЕНКА ОСТАТОЧНЫЙ ЗНАНИЙ ОБУЧАЮЩИХСЯ**

**2.1 Разработка базы данных**

База данных — совокупность данных, хранимых в соответствии со схемой данных, манипулирование которыми выполняют в соответствии с правилами средств моделирования данных.

Разработанная база данных содержит таблицы:

1. Тестеры (рис.4) – хранит в себе ФИО тестера, логин и пароль.

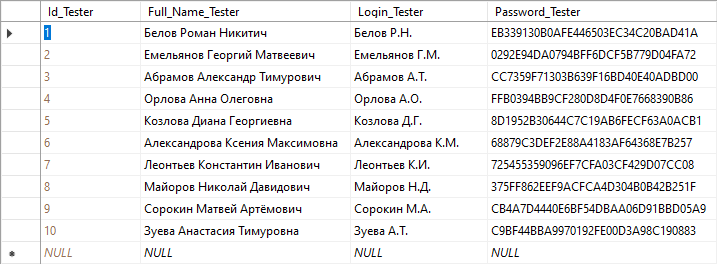


Рисунок 4 – Таблица «Тестеры»

1. Результаты (рис.5) – хранит в себе результаты прохождения тестов.

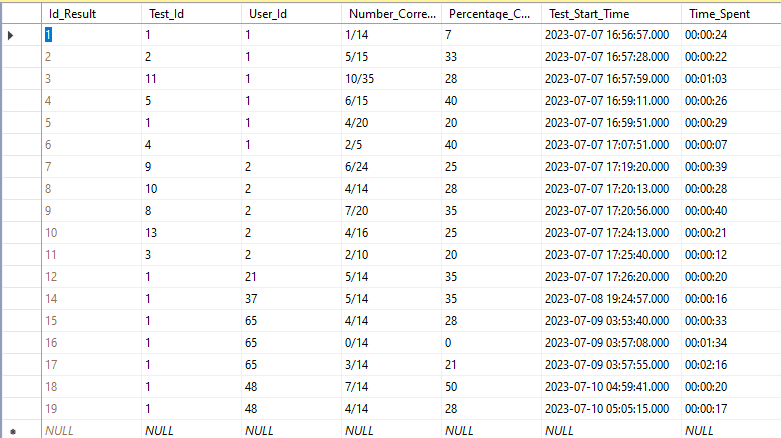


Рисунок 5 – Таблица «Результаты»

1. Вопросы (рис.6) – хранит в себе вопросы тестов.



Рисунок 6 – Таблица «Вопросы»

1. Группы (рис.7) – хранит в себе группы пользователей.

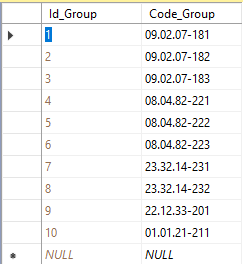


Рисунок 7 – Таблица «Группы»

1. Администратор (рис.8) – хранит в себе логин и пароль администратора.

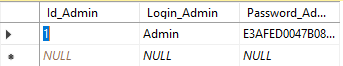


Рисунок 8 – Таблица «Администраторы»

1. Тесты (рис.9) – хранит в себе название теста и ограничение во времени для прохождения теста.

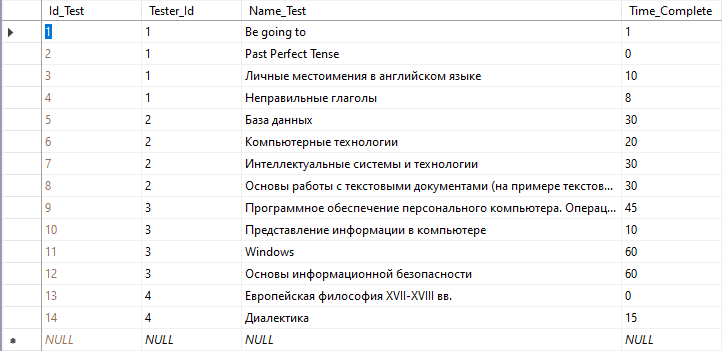


Рисунок 9 – Таблица «Тесты»

1. Пользователи (рис.10) – хранит в себе данные пользователей.

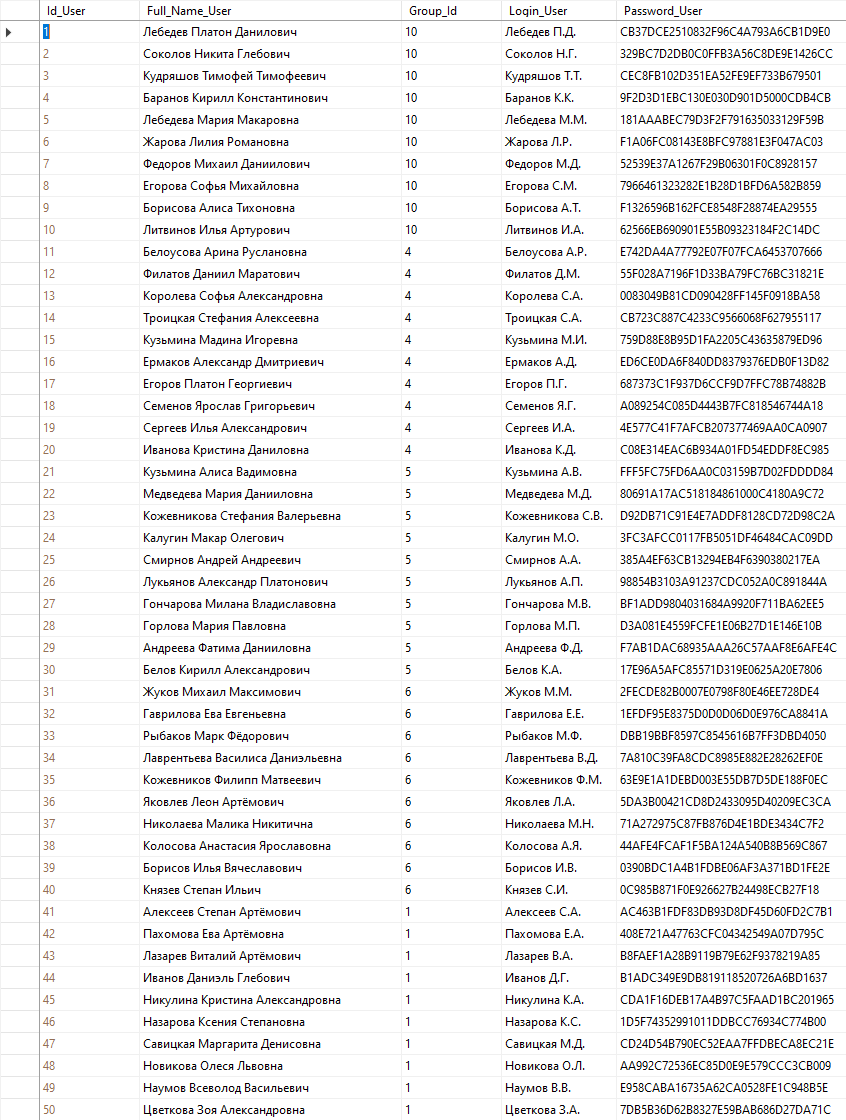


Рисунок 10 – Таблица «Пользователи»

После составления всех таблиц и внесения в них данных была построена диаграмма связей между таблицами (рис.11).

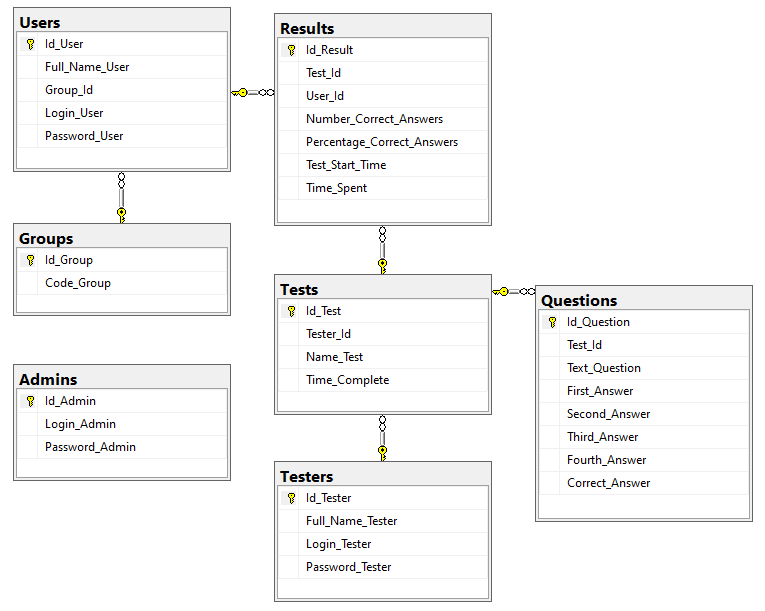


Рисунок 11 – Диаграмма

## **2.2 Разработка программного обеспечения**

Программное обеспечение (ПО) — программа или множество программ, используемых для управления компьютером (ISO/IEC 26514:2008).

После запуска ПО отображается «Главная форма» (рис.12). На данной форме есть возможность начать прохождение теста либо редактировать информацию в базе данных.

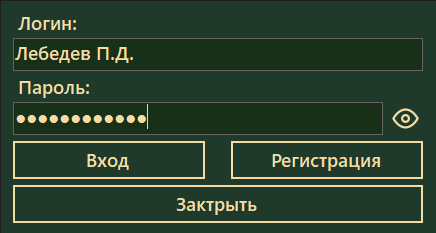


Рисунок 12 – Форма авторизации

Для регистрации аккаунта необходимо нажать кнопку «Регистрация» ввести всю необходимую информацию и выбрать тип аккаунта.

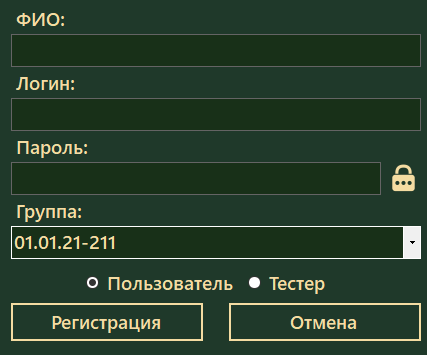


Рисунок 13 – Форма авторизации

В случае неверно введенного логина или пароля несколько раз открывается форма «Капча» (рис.13), которая защищает систему от взлома с помощью подбора логина и пароля. Для продолжения авторизации необходимо ввести в поле символы, показанные на изображении. При провале прохождения данного уровня защиты программа блокируется на определенный отрезок времени, который увеличивается с каждой проваленной попыткой. В случае верного ввода символов программа возвращает форму авторизации.

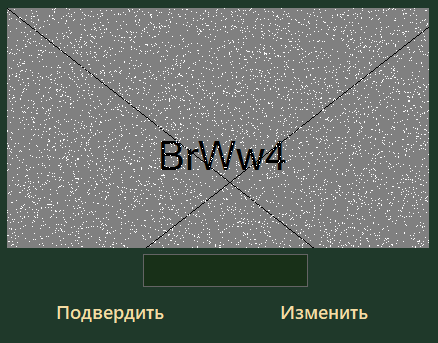


Рисунок 14 – Проверка на взлом программы

Если логин и пароль был введен корректно открывается форма «Меню пользователя» (рис.15), на которой можно начать прохождение теста либо посмотреть результаты.

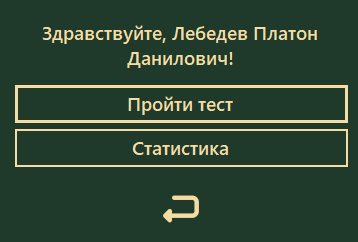


Рисунок 15 – Меню пользователя

При выборе прохождение теста открывается форма «Выбор теста» (рис.15), на которой можно выбрать тестера и тест для прохождения.

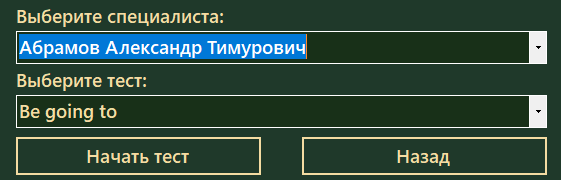


Рисунок 16 – Выбор теста

После выбора теста открывается форма для прохождения тестов (рис.16). На ней отображаются:

1. Название выбранного теста;
2. Текст вопроса;
3. Варианты ответов;
4. Номер вопроса и общее количество вопросов;
5. Кнопки для переключения вопросов.

По ознакомлению с вопросом необходимо выбрать один из 4 вариантов ответа, и перейти к следующему вопросу, в случае если пользователь забудет выбрать вариант ответа, ему высветится предупреждение, если вариант не будет выбран, по окончанию теста вопрос засчитается как не верный. Так же, если это было указано тестером, сверху отображается таймер, по окончанию которого, тест будет завершен, а все не отвеченные вопросы будут засчитаны как неверные. Вопросы и варианты ответов идут в рандомном порядке. Сделано это для усложнения списывания теста.

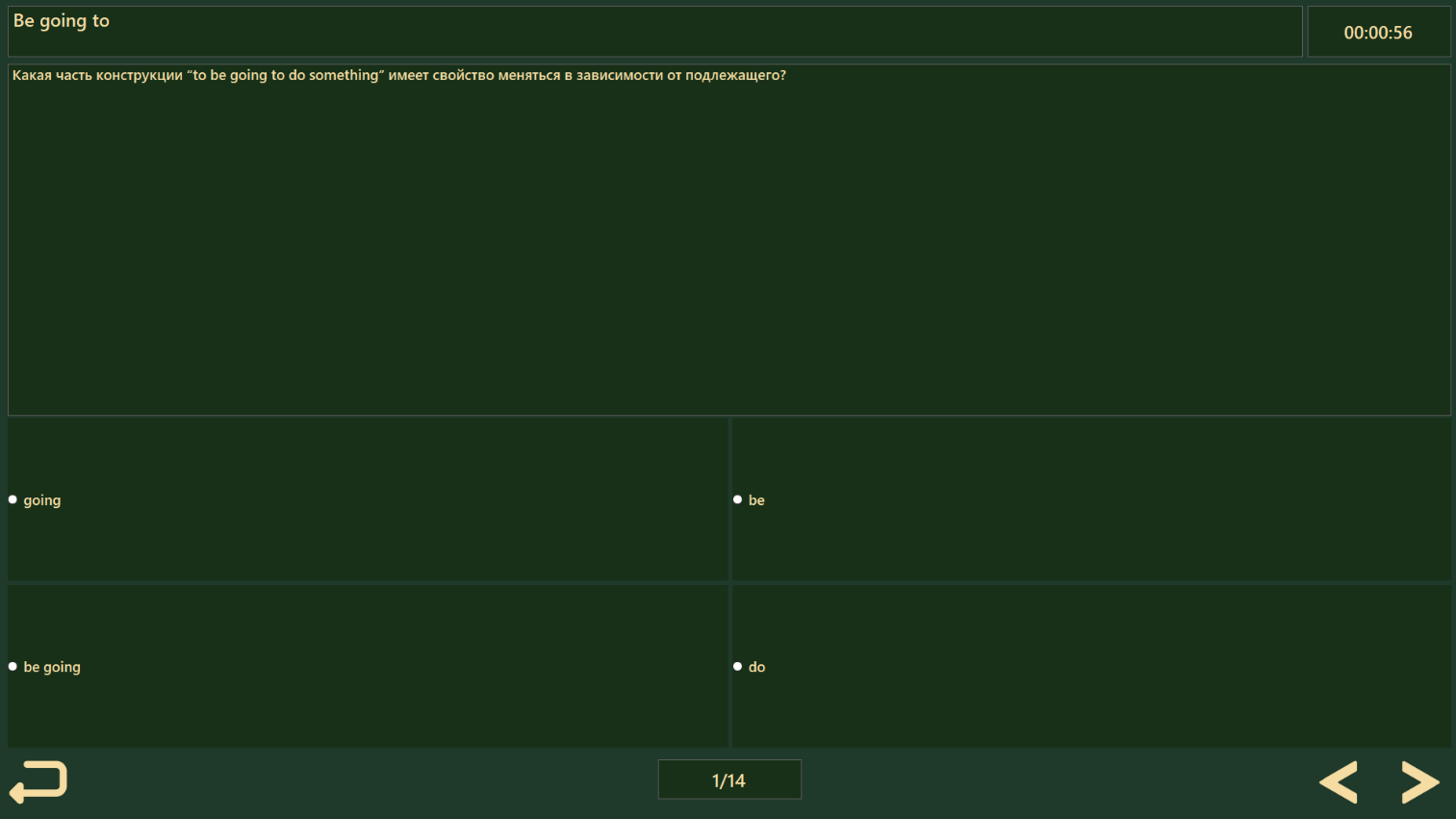


Рисунок 17 – Форма «Тестирование»

По окончанию тестирования появляется форма «Результат» (рис.18), отображающая полученную оценку и количество верных ответов из общего количество вопросов.



Рисунок 18 – Результат

Для просмотра оценок или редактирования данных на главной форме (Рис.13) необходимо нажать кнопку «Редактировать»

Так же результаты уже пройдённых тестов можно посмотреть на форме «Статистика», на ней можно увидеть данные теста, результат тестирования, дата прохождения теста, и время затраченое на прохождения.



Рисунок 19 – Статистика

Далее необходимо так же ввести логин и пароль, но на этот раз уже тестера. После авторизации отображается форма «Меню тестера» (рис.20), которая позволяет перемещаться между таблицами системы.

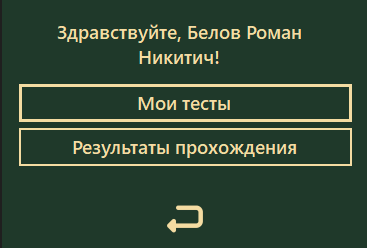


Рисунок 20 – Меню тестера

Кнопка «Мои тесты» переносит тестера на соответствующую форму (рис.21). На данной форме отображаются тесты данного тестера. Можно создать новый тест, редактировать уже созданный, или удалить тест и связанные с ним вопросы и результаты прохождения.

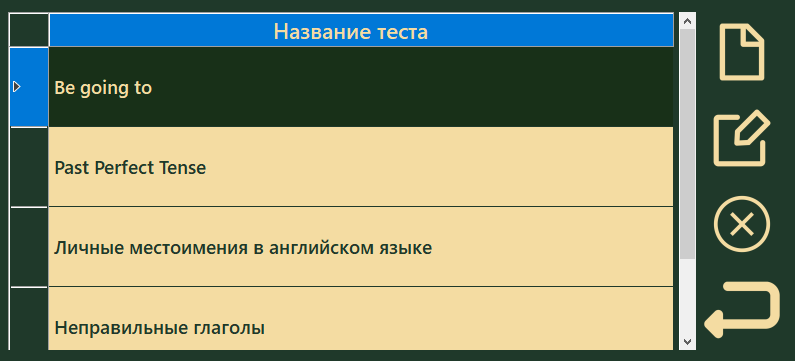


Рисунок 21 – Форма «Мои тесты»

При создании нового теста открывается специальная форма. Необходимо ввести название теста и время прохождения, если ограничений не планируется необходимо написать 0. Далее необходимо ввести текст вопроса, варианты ответа и выбрать верный вариант, после чело необходимо нажать кнопку «Добавить вопрос». Если вариант ответа не выбран или не все поля заполнены вопрос не добавится. Далее необходимо нажать на кнопку «Новый вопрос», форма очистится и можно будет внести новый пароль. В случае изменений необходимо нажать на кнопку изменить вопрос, если данный шаг будет пропущен, изменение не будет внесено. Для удаления вопроса необходимо нажать на соответствующую кнопку. По окончанию изменений необходимо нажать на кнопку «Сохранить изменения». Так же можно закрыть форму и изменения не будут сохранены.



Рисунок 22 – Форма «Новый тест»

Кнопка «Изменить тест» открывает редактор теста, выбранного в списке (рис.21). Данная форма отличается от предыдущей только наличием вопросов, все остальные шаги остаются без перемен.

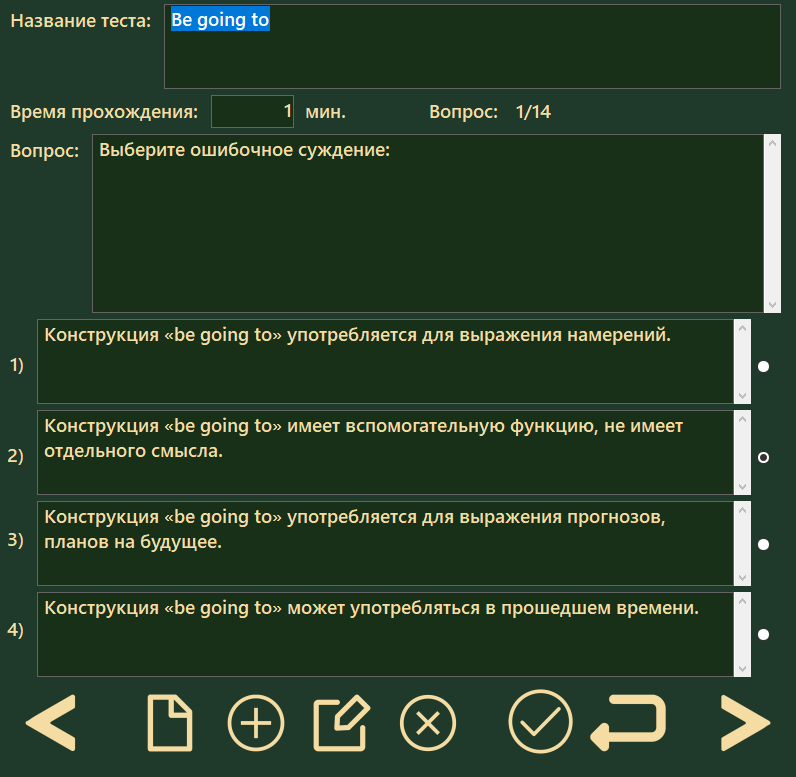


Рисунок 23 – Форма «Изменить тест»

Для входа в блок администратора необходимо ввести соответствующий логин и пароль, стандартными являются Admin Admin. Далее откроется меню, в котором можно выбрать форму для редактирования.

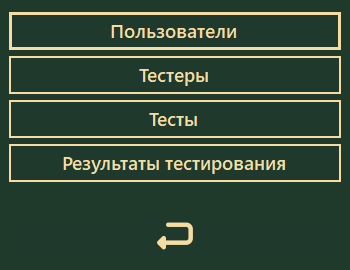


Рисунок 24 – Форма «Меню тестера»

Первой формой является «Пользователи» на ней можно увидеть список данных пользователей, количество записей, установить фильтр, и перейти на формы «Редактировать пользователей» и «Редактировать тесты».

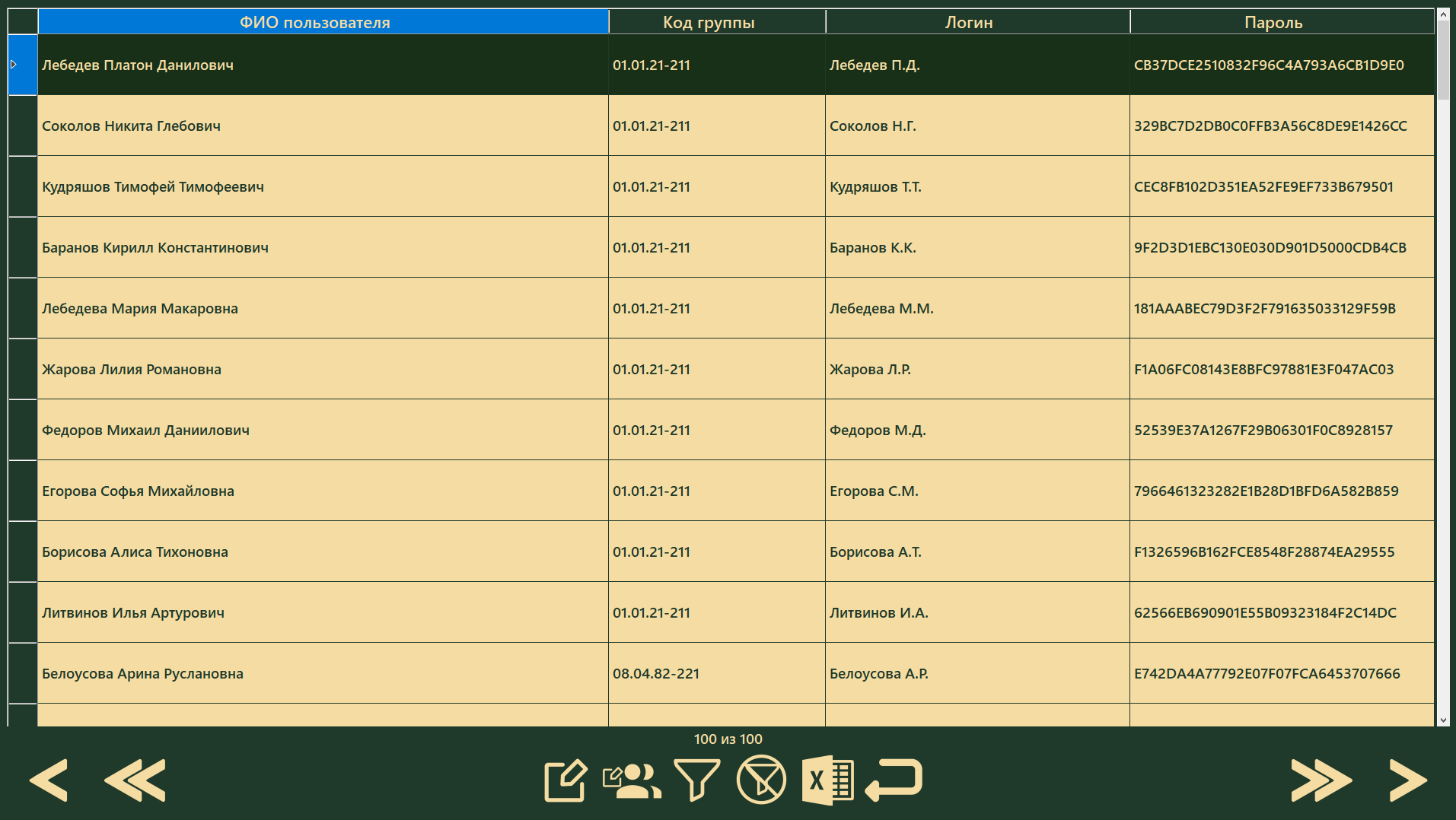


Рисунок 25 – Форма «Пользователи»

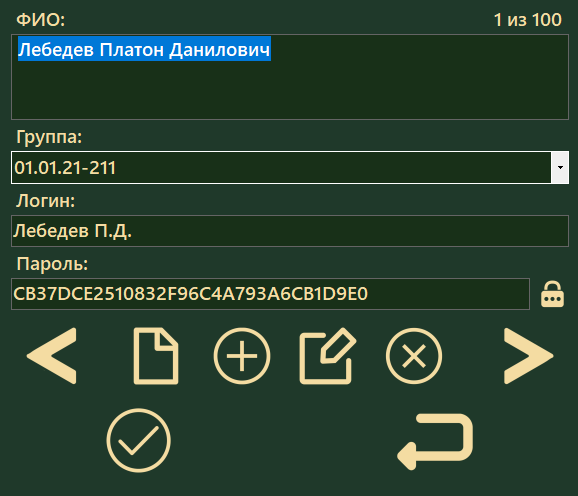


Рисунок 26 – Форма «Редактировать пользователей»



Рисунок 27 – Форма «Редактировать тесты»

На следующей форме можно увидеть информацию о тестерах. На ней так же можно отфильтровать данные, выгрузить их в Excel таблицу и редактировать тестеров.

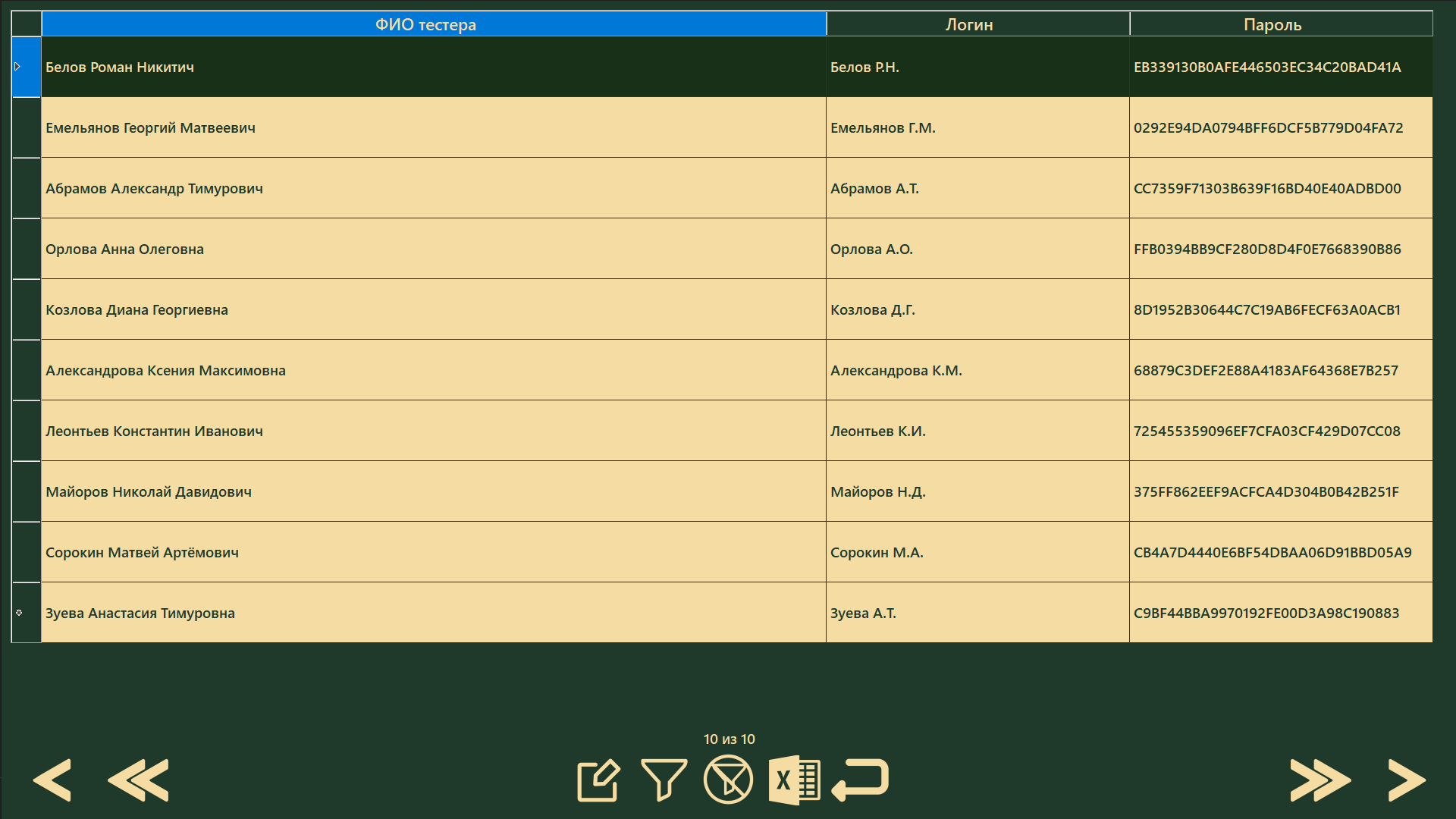


Рисунок 28 – Форма «Оценки»

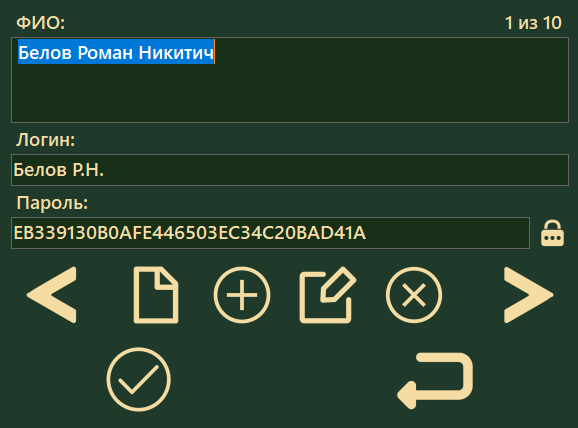


Рисунок 29 – Форма «Редактировать тестеров»

Далее идет форма «Тесты». На ней отображена информацию о всех сохранённых тестах. Администратор может только удалять тесты и выгружать их в таблицу Excel.

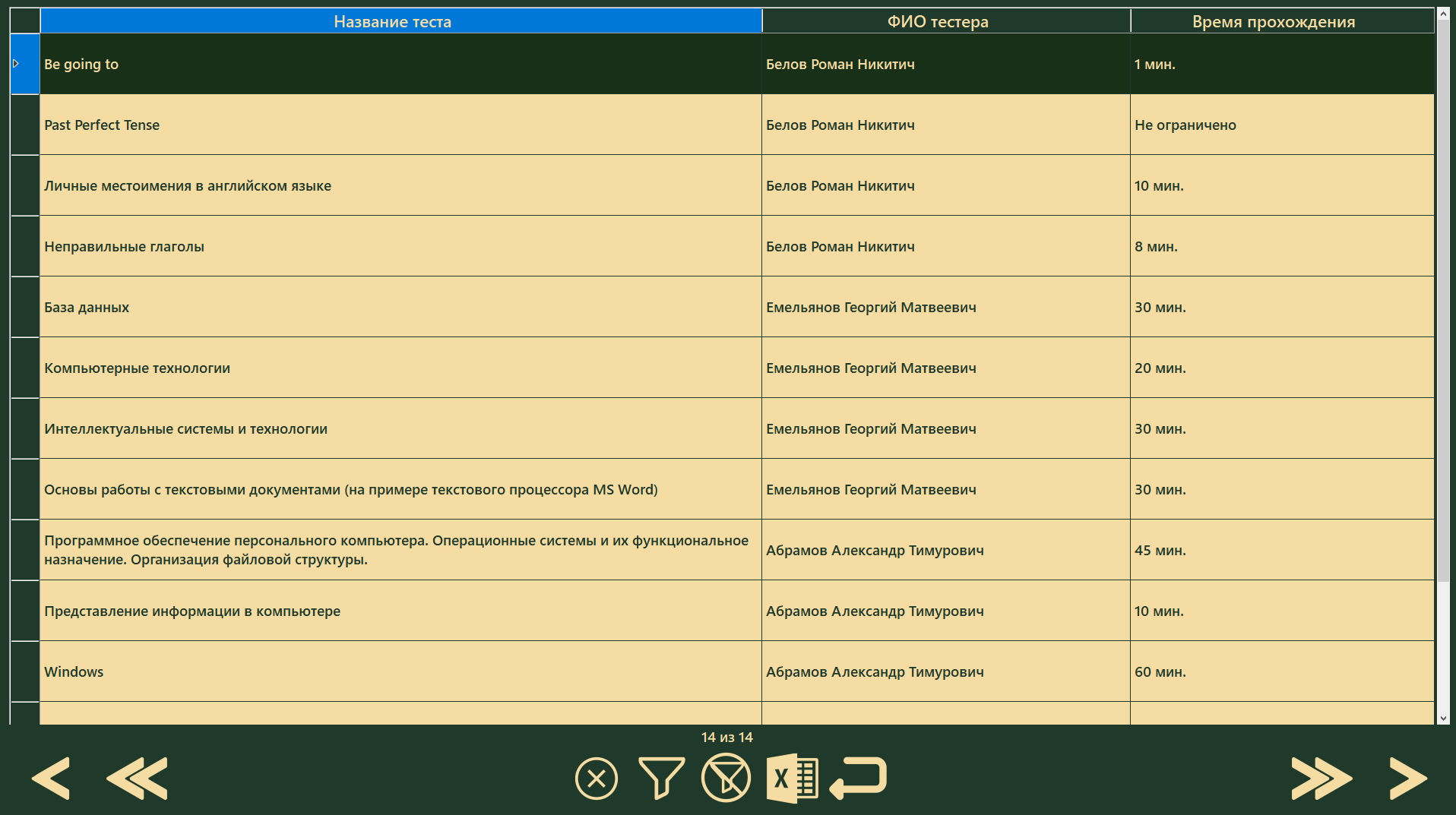


Рисунок 30 – Форма «Тесты»

Последней формой является «Результаты тестирования». На ней можно увидеть информацию о пройденных тестах

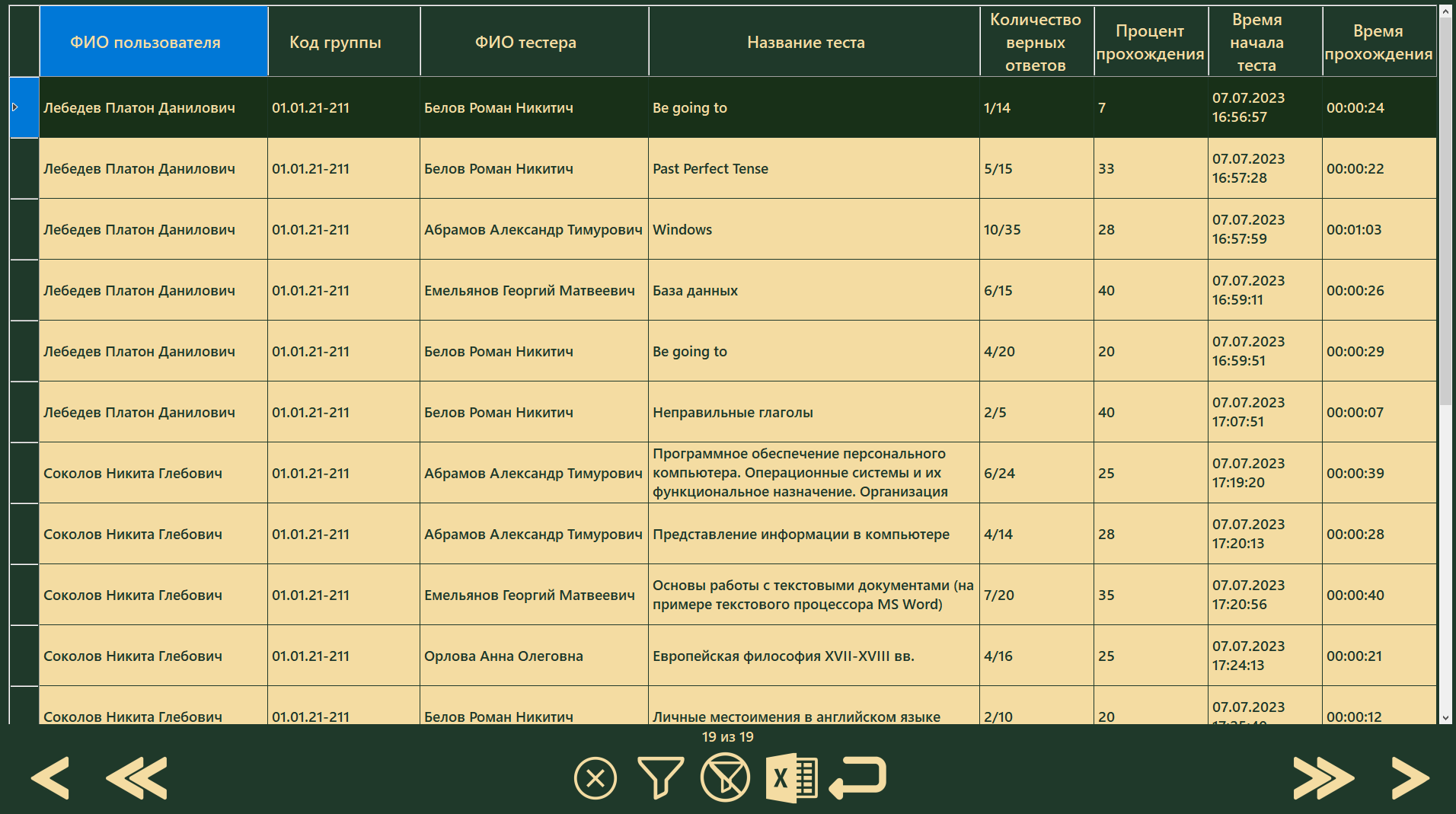


Рисунок 31 – Форма «Результаты тестирования»

**ЗАКЛЮЧЕНИЕ**

Достаточно широкое понимание информационной системы подразумевает, что её неотъемлемыми компонентами являются данные, техническое и программное обеспечение, а также персонал и организационное обеспечение. Широко трактует понятие «информационной системы» федеральный закон Российской Федерации «Об информации, информационных технологиях и о защите информации», подразумевая под информационной системой совокупность содержащейся в базах данных информации и обеспечивающих её обработку информационных технологий и технических средств.

База данных — совокупность данных, организованных в соответствии с концептуальной структурой, описывающей характеристики этих данных и взаимоотношения между ними, которая поддерживает одну или более областей применения.

Целью дипломной работы была разработка информационной системы, что позволит увеличить не только эффективность, но и качество при работе с информацией.

В результате выполнения дипломной работы были выполнены следующие задачи:

1. Изучена структура организации.
2. Разработан эскизный проект будущей систем.
3. Построена база данных.
4. Разработано пользовательское приложение.
5. Проведено тестирование системы и защиты базы данных.

Система разработана таким образом, что в случае необходимости будет возможность добавить новые системные блоки и расширить возможности программы. На данном этапе разработанное ПО соответствует выдвинутым требованиям.