

Министерство образования и молодежной политики Свердловской области  
Государственное автономное профессиональное образовательное учреждение  
Свердловской области  
«Камышловский педагогический колледж»

Дипломный проект

**Разработка автоматизированной информационной системы «Контроль  
состояния здоровья студентов ГАПОУ СО «Камышловский  
педагогический колледж» средствами Visual Studio C# и SQL Server**

09.02.07 Информационные системы и программирование

**Исполнитель:**  
Обоскалов Н.Д.,  
студент 4<sup>п</sup> группы  
**Руководитель:**  
Мясников С.В.,  
преподаватель

Камышлов, 2024

## Оглавление

<b>Пояснительная записка .....</b>	<b>3</b>
Введение.....	4
Описание предметной области, анализ и выбор методов решения .....	10
Описание информационной модели.....	10
Модель вариантов использования Use Case .....	11
Моделирование потоков данных (выделение бизнес-процессов).....	14
Определение требований будущих пользователей, целей, задач и функций программного продукта.....	19
Анализ существующих аналогов решения проекта .....	22
Разработка технического задания.....	24
Проектирование информационной системы .....	25
Проектирование базы данных.....	25
Логическое моделирование .....	26
Составление глоссариев и спецификаций .....	29
Создание физической модели базы данных .....	29
Использование импорта .....	32
Алгоритмы системы .....	
<b>Ошибка! Закладка не определена.</b>	
Создание SQL скрипта базы данных.....	34
Проектирование приложений конечных пользователей.....	35
Выбор среды разработки приложений для ИС .....	35
Проектирование с использованием UML диаграмм .....	36
Программная реализация информационной системы .....	40
Разработка интерфейса приложений.....	40
Многопользовательский режим.....	48
Реализация функций поиска и фильтрации данных .....	53
Реализация функций просмотра и генерации отчетов .....	55
Реализация дополнительных функций приложений .....	57
Тестирование приложений и отладка программных модулей .....	58
Создание инсталлятора программного обеспечения.....	64
Разработка сопровождающей технической документации .....	65

Оценка экономической/социальной эффективности проекта .....	66
<b>Заключение .....</b>	<b>76</b>
<b>Список литературы .....</b>	<b>78</b>
<b>Программный продукт «Автоматизированная информационная система «Контроль состояния здоровья студентов ГАПОУ СО «Камышловский педагогический колледж».....</b>	<b>82</b>
<b>Приложения</b>	

## **Пояснительная записка**

### **Введение**

В информационном обществе сегодня эффективность любой деятельности тесно связана с обработкой огромных объемов данных. Ручное управление информацией становится неэффективным и медленным. Внедрение передовых компьютерных технологий становится неотложной необходимостью. Для решения проблем обработки информации используются современные компьютеры с соответствующим программным обеспечением, системами управления базами данных [4].

В медицинских кабинетах колледжей часто требуется заполнение, хранение и печать большого количества отчетов и документов. Это занимает много времени и усилий при ручной обработке на бумаге, что снижает эффективность работы сотрудников. В связи с этим автоматизация информационных систем становится необходимостью для оптимизации процессов хранения и обработки данных.

В исследовании использовались следующие нормативно-правовые акты:

1. Федеральный закон «О персональных данных»;
2. Федеральный закон от 21.11.2011 N 323-ФЗ "Об основах охраны здоровья граждан в Российской Федерации";
3. Постановление Правительства Российской Федерации от 25.06.2007 N 397 "О порядке проведения профилактических медицинских осмотров";
4. Приказ Министерства здравоохранения Российской Федерации от 10.12.2012 N 1043н "Об утверждении Порядка проведения диспансеризации определенных групп взрослого населения";

5. Положение о медицинском кабинете ГАПОУ СО «Камышловский педагогический колледж»;

6. СанПиН 2.1.2.3684-21 "Санитарно-эпидемиологические требования к условиям осуществления образовательной деятельности в организациях среднего профессионального и высшего образования".

Актуальность разработки автоматизированной информационной системы заключается в структурировании необходимой информации, хранящийся в большом объёме [23]. Введение АИС позволит решить ряд проблем, с которыми сталкиваются при ручном ведении документации, такие как:

- возможность просмотра всех данных о здоровье студентов в одном месте, что облегчит поиск информации и ведение статистики;

- повышение эффективности обработки документов, так как система автоматически заполнит соответствующие бланки документов при вводе данных;

- уменьшение трудоемкости при оформлении отчетной документации, так как готовые документы можно немедленно распечатать или сохранить в электронном виде.

Таким образом, использование АИС в дальнейшем будет иметь ряд преимуществ перед ручным ведением документации, что позволит оптимизировать работу медицинского персонала и улучшить качество медицинского обслуживания студентов.

Учитывая вышеизложенное, можно выделить ряд противоречий:

- между необходимостью применения информационных технологий в решении вопросов автоматизированной обработки информационных потоков и отсутствием необходимой автоматизированной информационной системы в ГАПОУ СО «Камышловский педагогический колледж»;

— между недостаточной эффективностью сбора сведений о медицинских данных студентов и отсутствием автоматизированной информационной системы в ГАПОУ СО «Камышловский педагогический колледж».

Выявленные противоречия позволили определить проблему: необходимость автоматизации процесса контроля состояния здоровья студентов ГАПОУ СО «Камышловский педагогический колледж» средствами автоматизированных информационных систем (АИС).

Следовательно, объектом проектирования является процесс контроля состояния здоровья студентов в ГАПОУ СО «Камышловский педагогический колледж», включающий в себя сбор и обновление медицинских данных, проведение профилактических мероприятий, реагирование на медицинские проблемы, запрос и обработку медицинских документов, формирование отчетности и сотрудничество с медицинскими организациями.

Цель дипломного проекта: спроектировать и внедрить информационную систему «Контроль состояния здоровья студентов ГАПОУ СО «Камышловский педагогический колледж» средствами Visual Studio C# и SQL Server.

Для достижения поставленной цели необходимо решить следующие задачи:

1. Изучить и детально описать предметную область, в которой будет разрабатываться автоматизированная информационная система (АИС). Для этого необходимо провести тщательный сбор и анализ требований к АИС, определить ее функциональные возможности и особенности;
2. Определить требования будущей системы, цели, задачи и функции программного продукта. Для этого необходимо учесть все особенности и потребности конечных пользователей, а также разработать соответствующие формы, отчеты и диалоги.
3. Реализовать информационную систему средствами выбранной среды разработки. Выполнить проектирование базы данных, которая будет

хранить все необходимые медицинские данные о студентах. Определить ее логическую и физическую структуру, выбрать наиболее подходящую СУБД и разработать соответствующие таблицы и запросы.

4. Протестировать приложения и провести отладку программных модулей. Проверить все функциональные возможности АИС, выявить и устранить все возможные ошибки и недочеты, а также разработать сопровождающую техническую документацию.

5. Провести оценку экономической эффективности проекта. Учесть все затраты и выгоды от внедрения и эксплуатации АИС, а также определить ее потенциальный вклад в улучшение качества медицинского обслуживания студентов.

Для решения поставленных задач применялся комплекс взаимодополняющих теоретических и эмпирических методов исследования. К методам теоретического исследования, использованным в работе, можно отнести теоретический анализ научной и методической литературы, анализ профессионального опыта, обобщение, описание. Эти методы направлены на создание теоретических обобщений, установление и формулирование закономерностей изучаемого явления. К эмпирическим методам, использованным в работе, можно отнести метод беседы с лицами, работающими в медицинском кабинете, описание предметной области, моделирование различных схем по методологиям, тестирование приложения на различные ошибки.

При проектировании и разработке автоматизированной информационной системы (АИС) для хранения и обработки медицинских данных студентов была использована структурно-функциональная методология. Эта методология предусматривает разбиение системы на функциональные компоненты и описание их взаимодействия с помощью различных моделей.

В структурном анализе и проектировании АИС были использованы следующие распространенные модели:

— DFD (Data Flow Diagrams) - диаграммы потоков данных, которые описывают перемещение данных между функциональными процессами и внешними источниками информации;

— UML (Unified Modeling Language) - язык графического описания, который позволяет визуализировать структуру системы, ее компоненты и их взаимодействие;

— функциональная модель SADT (Structured Analysis and Design Technique) - метод структурированного анализа и проектирования, который предусматривает разбиение системы на иерархию функциональных блоков и описание их взаимодействия.

Новизна дипломной работы заключается в разработке АИС, специально адаптированной к потребностям образовательного учреждения, что учитывает специфические требования и особенности процессов контроля состояния здоровья студентов. Интеграции современных методов и технологий для эффективной обработки, и анализа медицинских данных, что позволяет повысить точность, доступность и безопасность информации.

Практическая значимость курсового проекта состоит в возможности проектирования информационной системы, которая позволит разгрузить сотрудника медицинского кабинета и использовать современные цифровые технологии для контроля состояния здоровья студентов. Исследование может представлять интерес для студентов, обучающихся по специальности 09.02.07 «информационные системы и программирование».

Данный дипломный проект состоит из основных разделов:

1. В первом разделе проводится анализ и описание предметной области, и выбор методов решения, моделируя потоки данных, определяются требования будущих пользователей, целей, задач и функций программного продукта, выполняется анализ существующих аналогов решения проекта и разработка технического задания.



2. Во втором разделе проводится проектирование информационной системы, в которое входит проектирование базы данных и приложений конечных пользователей. Проектирование базы данных представляет из себя создание логической модели, представление глоссариев и спецификаций, создание физической модели базы данных. Проектирование приложений конечных пользователей включает в себя выбор среды разработки приложения для АИС, проектирование с использованием UML диаграмм, определение функций программного продукта с применением методологии SADT и проектирование пользовательского интерфейса.

3. В третьем разделе приводится программная реализация информационной системы, в которую входит интерфейс приложения, функции поиска и фильтрации данных, функции просмотра и генерации отчётов, а также дополнительные функции приложения. Помимо вышеперечисленных подразделов описывается тестирование приложения и отладка программных модулей.

4. В четвертом разделе описывается разработка сопровождающей технической документации, а именно руководство пользователя.

5. В пятом разделе приводится обоснование методики расчета социальной эффективности результатов дипломного проектирования, расчет показателей социальной эффективности.

## **Описание предметной области, анализ и выбор методов решения**

### **Описание информационной модели**

Анализ предметной области, позволяет выделить ее сущности, определить границы проекта и первоначальные требования к функциональности. Модель предметной области должна быть документирована, храниться и поддерживаться в актуальном состоянии до этапа реализации. Для документирования могут быть использованы различные средства [1].

Описание предметной области: в медицинском кабинете ГАПОУ СО «Камышловский педагогический колледж» происходит контроль состояния здоровья студентов. Общей целью данной предметной области является обеспечение безопасности и заботы о здоровье студентов, а также оперативное выявление и управление медицинскими проблемами, способными повлиять на их обучение и общее благополучие. Предметная область контроля состояния здоровья студентов охватывает процессы мониторинга, учета и обеспечения безопасности здоровья студентов в образовательном учреждении. Эта область включает в себя сбор и анализ информации о физическом состоянии студентов, хронических заболеваниях, аллергиях и других медицинских параметрах.

Информационная модель — модель объекта, представленная в виде информации, описывающей существенные для данного рассмотрения параметры и переменные величины объекта, связи между ними, входы и выходы объекта и позволяющая путём подачи на модель информации об изменениях входных величин моделировать возможные состояния объекта.

Описание информационной модели: медицинский работник в устной форме запрашивает документы о состоянии здоровья у студента или ответственного за посещаемость, по электронной почте или запросом в виде

документа у медицинской организации или прошлого учебного заведения. Ответственный за посещаемость сдает в виде документа справку о пропуске. Справку о пропуске по болезни в виде документа студент сдает ответственному за посещаемость. Студент наблюдается или наблюдался в медицинской организации что подтверждается документами (см. Рисунок 1).

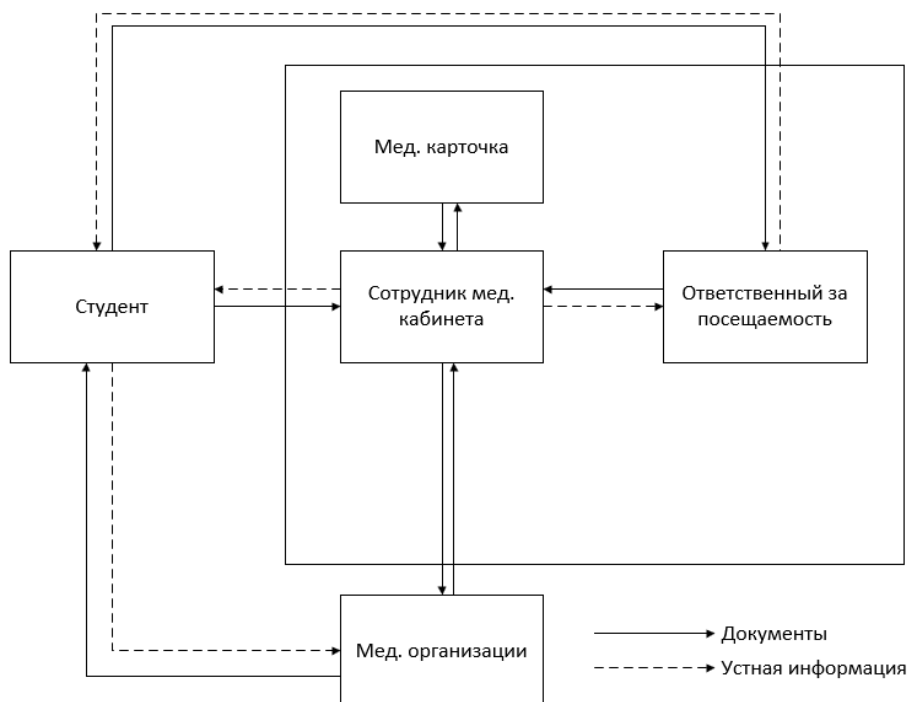


Рисунок 1 - Информационная модель ПО

### Модель вариантов использования Use Case

После построения информационной модели была построена диаграмма вариантов использования (use case diagram) в UML (Unified Modeling Language), которая отражает отношения между актерами и прецедентами. Данная диаграмма является составной частью модели прецедентов, позволяющая описать систему на концептуальном уровне [20].

В качестве подсистемы для построения диаграммы вариантов использования была выбрана система «Контроль состояния здоровья

студентов колледжа». Далее были описаны основные объекты: актёры, прецеденты и связи между ними (см. Рисунок 2).

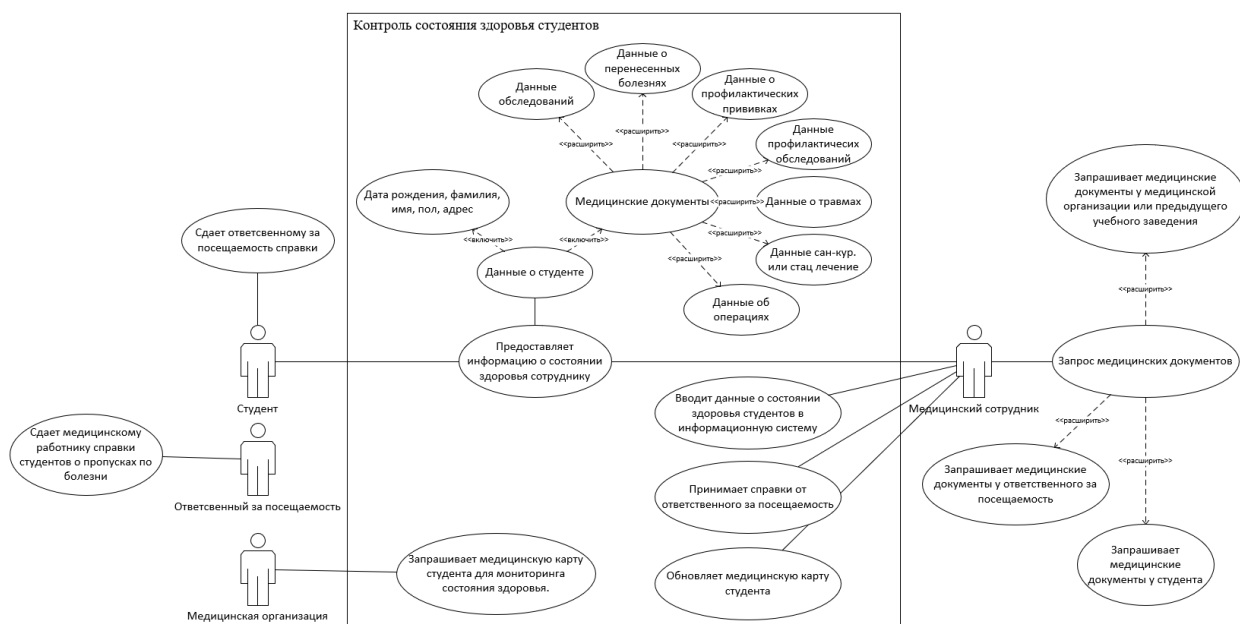


Рисунок 2 - Диаграмма вариантов использования

Описание диаграммы вариантов использования: диаграмма представляет систему контроля состояния здоровья студентов. На ней изображены взаимодействия между различными участниками процесса и поток информации.

Основные группы пользователей системы:

- медицинский работник колледжа. Основной пользователь системы, ответственный за мониторинг состояния здоровья студентов и ведение медицинских данных;
- студент. Пользователь, чье здоровье отслеживается в системе. Предоставляет медицинские данные и документы;
- ответственный за посещаемость в группе. Пользователь, отвечающий за учет посещаемости студентов в группе и сдачу медицинских справок;
- медицинская организация. Внешний пользователь, имеющий интерес к медицинским данным студентов.

Варианты использования:

— отслеживание состояния здоровья студентов. Описание: медицинский работник вносит и обновляет данные о состоянии здоровья студентов в информационной системе. Связи: связан с актерами "Медицинский работник колледжа" и "Студент";

— запрос медицинских документов. Описание: медицинский работник запрашивает медицинские документы у студентов, ответственного за посещаемость и медицинских организаций. Связи: связан с актерами "Медицинский работник колледжа", "Студент", "Ответственный за посещаемость в группе" и "Медицинская организация";

— отчетность о пропусках по болезни. Описание: ответственный за посещаемость сдает медицинскому работнику справки о пропусках занятий по болезни студентов. Связи: связан с актерами "Ответственный за посещаемость в группе" и "Медицинский работник колледжа";

— информационная система. Описание: система для ввода, хранения и обновления данных о состоянии здоровья студентов. Связи: связан с актером "Медицинский работник колледжа";

— сотрудничество с медицинскими организациями. Описание: медицинский работник обеспечивает сотрудничество с внешними медицинскими организациями для обмена информацией о состоянии здоровья студентов. Связи: связан с актерами "Медицинский работник колледжа" и "Медицинская организация".

Отношения между актерами и прецедентами:

— медицинский работник колледжа. Взаимодействует с прецедентами "Отслеживание состояния здоровья студентов", "Запрос медицинских документов", "Отчетность о пропусках по болезни", "Информационная система" и "Сотрудничество с медицинскими организациями".

— студент. Взаимодействует с прецедентами "Отслеживание состояния здоровья студентов" и "Запрос медицинских документов".

— ответственный за посещаемость в группе. Взаимодействует с прецедентом "Отчетность о пропусках по болезни".

— медицинская организация. Взаимодействует с прецедентами "Запрос медицинских документов" и "Сотрудничество с медицинскими организациями".

### Моделирование потоков данных (выделение бизнес-процессов)

DFD (Диаграмма Поток Данных) позволяет визуализировать потоки данных в системе и их обработку, идентифицируя основные процессы, хранилища данных и внешние сущности. Это упрощает анализ и проектирование систем, фокусируя внимание на передаче, обработке и хранении информации внутри системы [2].

Диаграмма потоков данных - один из основных инструментов структурного анализа и проектирования информационных систем, существовавших до широкого распространения UML [1].

Была построена DFD диаграмма, при помощи приложения Microsoft Visio и реализована версия DFD модели до реинжиниринга (см. Рисунок 3).



Рисунок 3 - Бизнес-процесс «Формирование медицинской карточки»

Диаграмма описывает бизнес-процесс сбора и обработки данных о травмах студентов. В этом процессе участвуют студент и сотрудник медицинского кабинета.

Основные шаги бизнес-процесса:

1. Заполнение формы данными о травме (1.1). Участник: студент. Действие: студент заполняет форму с данными о своей травме. Результат: заполненная форма с данными о травме передается сотруднику медицинского кабинета.

2. Обновление информации о травмах студентов (1.2). Участник: сотрудник медицинского кабинета. Действие: сотрудник медицинского кабинета обновляет информацию о травме студента в системе на основе данных, предоставленных студентом. Результат: обновленная информация о травмах студентов.

3. Формирование отчета о травмах студента (1.3). Участник: Сотрудник медицинского кабинета. Действие: сотрудник медицинского кабинета формирует отчет о травмах студента. Результат: отчет о травмах студентов возвращается студенту.

На каждом этапе процесса происходит передача и обработка данных о травмах студента, что обеспечивает актуальность и полноту информации о состоянии здоровья студентов в образовательной организации.

Для того что бы построить модель после реинжиниринга была проведена оценка по количеству критических факторов успеха (КФУ). Критические факторы успеха – это те стратегические задачи, конкурентные возможности, результаты деятельности, которые каждая компания должна обеспечивать или стремиться к этому, чтобы быть конкурентоспособной и добиться успеха на рынке [37].

Далее были определены факторы для оценки: скорость, качество, стоимость, удобство обслуживания. По каждому бизнес-процессу выставлена оценка выполнения. Средний балл был рассчитан по формуле:  $S_i = (Z_{i1} * F_1 + Z_{i2} * F_2 + \dots + Z_{in} * F_n) / n$  где  $Z_{i1}, Z_{i2}, \dots, Z_{in}$  - оценки  $i$ -го бизнес-процесса

по каждому критическому фактору успеха,  $F_1, F_2, \dots, F_n$  - веса критических факторов успеха,  $n$  - количество критических факторов успеха.

Представлены результаты оценки бизнес-процессов по критическим факторам (см. Таблица 1) [9].

Таблица 1 - Сопоставление бизнес-процессов с критическими факторами успеха

<b>Бизнес - процесс \ КФУ</b>	<b>Скорость обработки информации</b>	<b>Качество информации</b>	<b>Удобство обслуживания</b>	<b>Средний балл процесса</b>
Заполнить форму данными о травме	2 (0,3)	4 (0,3)	2 (0,4)	2,6
Заполнить информацию о травмах студентов	3 (0,3)	3 (0,3)	2 (0,4)	2,6
Сформировать отчет о травмах студента	2 (0,4)	4 (0,3)	3 (0,3)	2,6
Общая оценка				2,6

Следующим этапом была разработка модели DFD модель после реинжиниринга.

Реинжиниринг обычно представляют, как фундаментальное переосмысление и радикальную перестройку бизнес-процессов в целях улучшения таких важных показателей, как стоимость, качество, уровень сервиса, скорость функционирования, финансы, маркетинг, построение информационных систем для достижения радикального, скачкообразного улучшения деятельности организации. Главной целью бизнес-реинжиниринга является резкое ускорение реакции предприятия на изменения в требованиях потребителей при многократном снижении затрат всех видов [5].

К предыдущей модели добавлены данные, которые указываются при каждом этапе. Данные хранят информации, которую облегчит поиск и работу.



По данным диаграмм сделал вывод, что автоматизация данного бизнес-процесса выгодна тем, что благодаря ей работа улучшится так как в базе данных храниться информация и упростит поиск и хранение.

После проведение реинжиниринга определил какую информацию необходимо хранить в базу данных для улучшения работы. (см. Рисунок 4).

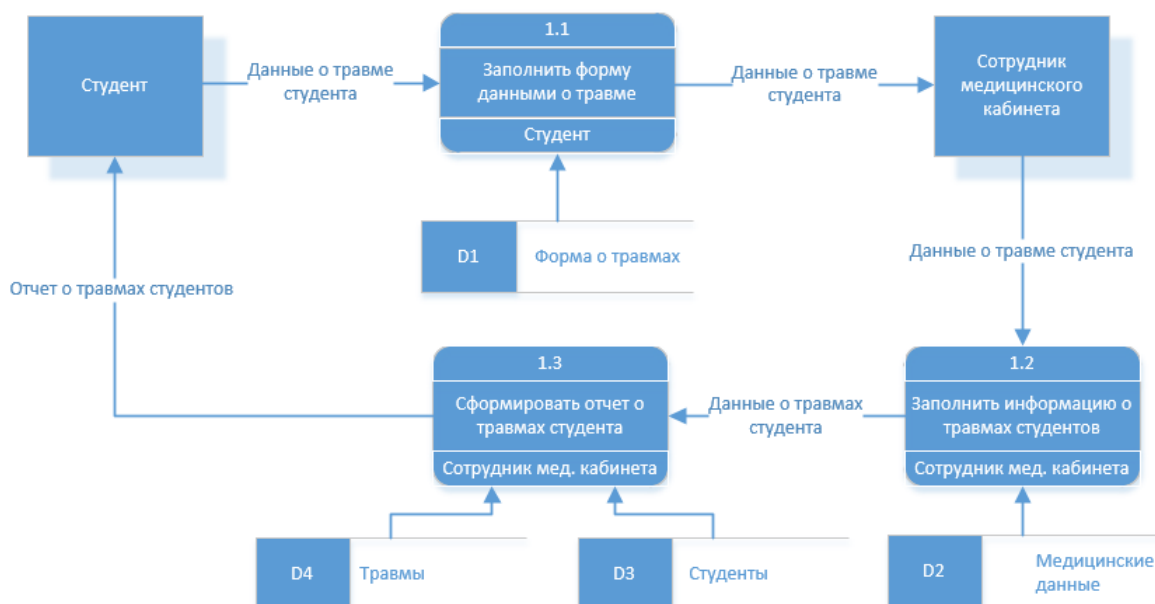


Рисунок 4 - Бизнес-процесс «Формирование медицинской карточки» после реинжиниринга

Описание процесса:

1. Заполнение формы данными о травме (1.1). Студент заполняет форму данными о травме. Вводимые данные: данные о травме студента. Выходные данные: заполненная форма о травме (D1);
2. Обновление информации о травмах студентов (1.2). Сотрудник медицинского кабинета обновляет информацию о травмах студентов на основе данных, предоставленных студентом. Вводимые данные: данные о травме студента. Выходные данные: обновленные медицинские данные (D2).
3. Формирование отчета о травмах студента (1.3). Сотрудник медицинского кабинета формирует отчет о травмах студента. Вводимые данные: данные о травмах студента (D3). Выходные данные: отчет о травмах студентов (D4). Студент получает отчет о травмах студентов после его

формирования сотрудником медицинского кабинета. Вводимые данные: отчет о травмах студентов (D4).

Данные, задействованные в процессе:

- D1: форма о травмах – содержит информацию, которую студент заполняет о своей травме;
- D2: медицинские данные – информация о травмах студентов, обновляемая сотрудником медицинского кабинета;
- D3: студенты – перечень студентов, для которых собираются данные о травмах;
- D4: травмы – отчет, формируемый на основе данных о травмах студентов.

Потоки данных:

- данные о травме студента поступают от студента к сотруднику медицинского кабинета через заполненную форму;
- сотрудник медицинского кабинета обновляет информацию о травме и формирует отчет, который затем передается обратно студенту.

Этот процесс обеспечивает своевременное обновление и точность данных о травмах студентов, что важно для мониторинга состояния их здоровья в образовательном учреждении.

По данным диаграмм сделан вывод, что автоматизация данного бизнес-процесса выгодна тем, что благодаря ей работа улучшится так как в базе данных храниться информация и упростит поиск и хранение. Представлены критические факторы успеха после реинжиниринга (см. Таблица 2).

Таблица 2 - Сопоставление бизнес-процессов с критическими факторами успеха после реинжиниринга

<b>Бизнес - процесс</b>	<b>КФУ</b>	<b>Скорость обработки информации</b>	<b>Качество информации</b>	<b>Удобство обслуживания</b>	<b>Средний балл процесса</b>
Заполнить форму данными о травме		5 (0,3)	5 (0,3)	4 (0,4)	4,6

Заполнить информацию о травмах студентов	4 (0,3)	5 (0,3)	5 (0,4)	4,6
Сформировать отчет о травмах студента	5 (0,4)	4 (0,3)	5 (0,3)	4,6
Общая оценка				4,6

## Определение требований будущих пользователей, целей, задач и функций программного продукта

Для определения требований будущих пользователей и функций программного продукта была построена SADT модель (см. Рисунок 5).

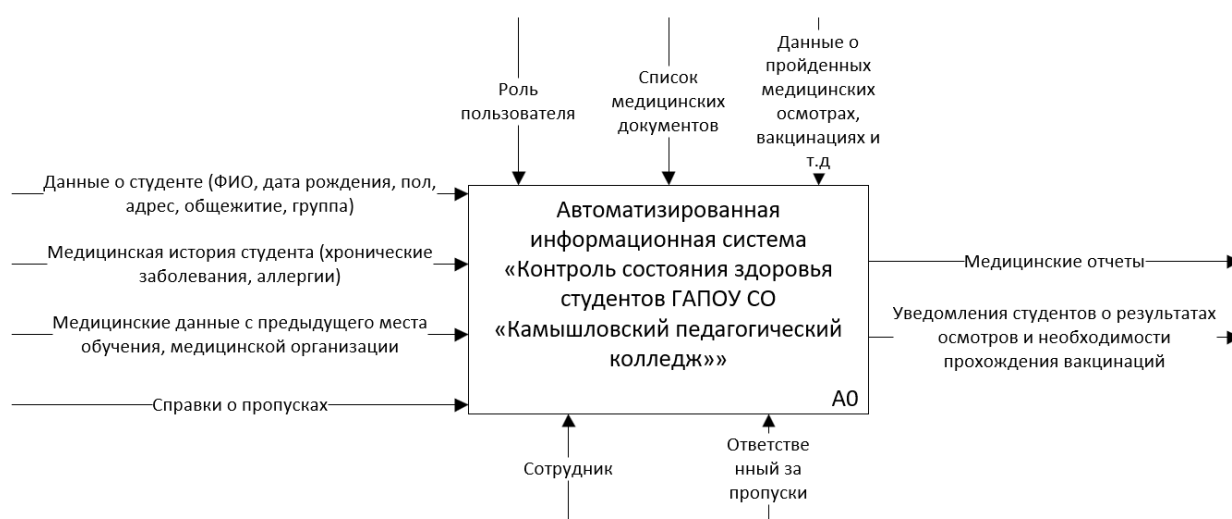


Рисунок 5 - Модель нулевого уровня методологии SADT

SADT (Structured Analysis and Design Technique) используется для моделирования структуры и функций программного продукта. В случае автоматизированной информационной системы «Контроль состояния здоровья студентов ГАПОУ СО «Камышловский педагогический колледж», применение SADT может быть представлено следующим образом:

Входящие данные:

- данные о студенте (ФИО, дата рождения, пол, адрес, общежитие, группа);
- медицинская история студента (хронические заболевания, аллергии);
- медицинские данные с предыдущего места обучения, медицинской организации;
- справки о пропусках.

Выходные данные:

- медицинские отчеты;
- уведомления студентов о результатах осмотров и необходимости прохождения вакцинаций.

Управление:

- роль пользователя;
- список медицинских документов;
- данные о пройденных медицинских осмотрах, вакцинациях и т.д.

Механизм:

- сотрудник;
- ответственный за пропуски.

Из SADT диаграммы SADT A0 можно определить следующие требования пользователей:

- медицинский персонал должен иметь возможность быстрого и удобного доступа к медицинским данным студентов; функционал для ввода, редактирования и удаления медицинской информации; возможность генерации отчетов о состоянии здоровья студентов и пропусках по болезни; доступ к истории медицинских данных студентов для анализа и управления;
- студент должен иметь возможность предоставления медицинских справок и документов; доступ к своим медицинским данным и истории посещений медицинского кабинета; возможность просмотра графика проведения профилактических мероприятий;

Цели программного продукта:

- оптимизация процессов учета и контроля состояния здоровья студентов. Обеспечение централизованного доступа к медицинской информации, сокращение времени на административные процедуры и повышение эффективности работы медицинского персонала;

- повышение качества медицинского обслуживания.

Предоставление инструментов для анализа и мониторинга состояния здоровья студентов, что позволит оперативно выявлять проблемы и предоставлять качественную медицинскую помощь.

Задачи программного продукта:

- создание базы данных медицинских записей студентов.

Разработка функционала для хранения и обработки информации о физическом состоянии, хронических заболеваниях, аллергиях и других медицинских параметрах студентов;

- разработка интерфейса пользователя. Создание удобного и интуитивно понятного пользовательского интерфейса для всех категорий пользователей с учетом их потребностей и привычек;

- реализация функционала формирования отчетов и аналитики.

Разработка возможности генерации отчетов о состоянии здоровья студентов, пропусках по болезни, а также инструментов для анализа медицинских данных.

Функции программного продукта:

- учет медицинских данных студентов. Ввод, хранение и обновление информации о состоянии здоровья студентов, включая результаты осмотров, анализов и консультаций;

- генерация отчетов и аналитика. Формирование различных отчетов о состоянии здоровья.

## Анализ существующих аналогов решения проекта

В ходе разработки создания приложения, были проанализированы 2 аналога приложения: MEDESK и eClinic. Проанализировав существующие аналоги разрабатываемой ИС, основные возможности были описаны, их плюсы и минусы были изложены, в итоге был сделан вывод.

### 1. MEDESK

Облачная информационная система, предназначенная для учета, управления и анализа медицинских данных врачей и пациентов (см. Рисунок 6). Создана для удобного доступа к медицинской информации в любое время и место.

The screenshot displays the MEDESK web application interface. On the left is a sidebar with navigation icons for 'Рабочий стол' (Dashboard), 'Расписание' (Schedule), 'Пациенты' (Patients), 'Организации' (Organizations), 'Статистика' (Statistics), 'Настройки' (Settings), and 'Помощь' (Help). The main area is titled 'Статистика' (Statistics) and shows a report for 'Сводка по закрытым лечением с 1-го по 31 марта 2015 г.' (Summary of closed treatments from March 1st to 31st, 2015). The report includes search filters for date, period, and type of selection, as well as output format and column selection options. The data is presented in a table with columns for ID, Date, Registrar, Treatment, Patient, and Payment amounts. The total sum for the period is 212777 p.

#	Открыто	Регистратор	Лечение	Пациент	К оплате	Итого
18 марта 2015 г.						
11	15.02.2015 10:59		Check Up 'Женское здоровье'	Симонова Анна	147636 p.	205297 p.
1	10.12.2014 17:52		Амбулаторный прием	Скалов Иван	6556 p.	6556 p.
4	05.02.2015 12:02		Check Up 'Мужское здоровье'	Петренко Петр Петрович	0 p.	0 p.
5	18.03.2015 13:19	Ангархаева Н.	Консультация врача	Иванов Иван	10330 p.	20520 p.
6	18.03.2015 13:16	Ангархаева Н.	Амбулаторный прием	Карпова Ирина	6000 p.	6000 p.
7	18.03.2015 13:20	Ангархаева Н.	Маммограф плановый	Иванов Иван	10482 p.	10482 p.
8	18.03.2015 12:54	Ангархаева Н.	Check Up 'Мужское здоровье'	Петренко Иван Артемьевич	1650 p.	1650 p.
9	12.12.2014 15:02		Check Up 'Женское здоровье'	Иванов Иван Иванович	1566 p.	1566 p.
10	24.11.2014 05:04		Тестовое обследование	Лискутов Петр	0 p.	0 p.
3	18.03.2015 12:48	Ангархаева Н.	Check Up 'Женское здоровье'	Алиева Лейла	0 p.	0 p.
Итого на сумму					212777 p.	

Рисунок 6 - Программа MEDESK

Основные возможности:

- ведение электронной медицинской карты пациента;
- учет и анализ медицинских данных и результатов лабораторных исследований.

Преимущества:

- удобный интерфейс;
- высокий уровень безопасности данных;

— возможность работы через веб-интерфейс или мобильное приложение.

Недостатки:

— дорогой тарифный план;  
— не всегда удобный поиск и доступ к необходимой информации в больших базах данных.

Лицензия: платная.

## 2. eClinic

Программа, позволяющая управлять графиком работы врачей, записывать пациентов на прием, информировать клиентов о записях - система отправит sms талон и напоминание о предстоящем приеме (см. Рисунок 7).  
Расширять список клиентов за счет записи с сайта клиники.

The screenshot displays the ECLINIC web interface for patient registration and appointment management. The interface is organized into several sections:

- Header:** ECLINIC – "ЕКЛИНИК" with navigation tabs: Пациенты, Регистрация, Задачи, Отчеты, Настройки.
- Patient Information:** Fields for Patient (Пациент), Payer (Платательщик), Specialist (Специалист), Date (Дата), Time (Время), Duration (Длительность), and Room (Кабинет). A dropdown for "Доп. информация" is also present.
- Diagnosis:** A text area for "Диагнозы МКБ" containing "к 111.9 Гипертензивная [гипертензивская] болезнь с преимущественным поражением сердца без длительной сердечной недостаточности".
- Services and Materials:** Tabs for "Услуги и материалы", "Медицинские документы", and "Телемедицина". The "Услуги" section includes a search bar and a table of services.
- Services Table:**

Услуга	Специалист	Дата	Цена за ед. руб.	Кол-во	Стоимость руб.
44681 Консультация врача категории "С"			2100	1	2100

Стоимость услуг без скидки, руб: 2100
- Materials:** A section for "Материалы" with a search bar and a table of materials.
- Materials Table:**

Название	Партия	Кол-во	Цена	Стоимость
Материалы не добавлены				

Стоимость материалов, руб: 0
- Payment Summary:** A section titled "Счет ОПЛАЧЕН" showing a breakdown of costs and payment details.
- Payment Summary Table:**

Дата оплаты	Тип оплаты	Сумма оплаты руб.
12.04.2023	Банковская карта	2100

Общая сумма оплат, руб: 2100

Рисунок 7 - Программа ECLINIC

Основные возможности:

— учет пациентов и их общей медицинской истории;

— учет всех клинических документов пациентов и документооборота.

Преимущества:

— понятный интерфейс со сложной организацией данных;  
— подключение государственных центров обмена медицинской информации.

Недостатки:

— высокая стоимость системы;  
— ограниченные варианты графических отчетов;  
— не всегда удобный интерфейс.

Лицензия: платная.

Вывод: разрабатываемая информационная система предоставляет значительные преимущества по сравнению с аналогами благодаря гарантированной безопасности данных, удобству использования и повышенной эффективности работы пользователей.

## **Разработка технического задания**

Техническое задание – исходный документ на проектирование технического объекта (изделия).

При разработке всей документации использовалось ГОСТ 19.105-78 Единая система программной документации (ЕСПД) [7]. В соответствии для разработки программного обеспечения разработала техническое задание с соответствием с ГОСТ. Техническое задание составлено с ГОСТ 19.201-78 (01.07.2023) [12]. Так как настоящий стандарт устанавливает порядок построения и оформления технического задания на разработку программы или программного изделия для вычислительных машин, комплексов и систем независимо от их назначения и области применения. Представлены разделы описывающие и подходящие для разработки приложения.



Техническое задание устанавливает:

- основное назначение разрабатываемого объекта;
- его технические характеристики;
- показатели качества и технико-экономические требования;
- предписание по выполнению необходимых стадий создания документации (конструкторской, технологической, программной и т. д.) и её состав;
- специальные требования.

Техническое задание содержит следующие разделы:

1. Введение.
2. Основания для разработки.
3. Назначение разработки.
4. Требования к программе или программному изделию.
5. Требования к программной документации.
6. Техничко-экономические показатели.
7. Стадии и этапы разработки.
8. Порядок контроля и приемки.
9. Приложения.

## **Проектирование программного продукта**

### **Проектирование базы данных**

Проектирование базы данных представляет собой важный этап, включающий создание комплекса взаимосвязанных моделей данных, где каждая модель отражает определенные аспекты предметной области. Этот процесс тесно связан с обработкой и манипулированием данными, поскольку

целью базы данных является хранение, организация и эффективное использование информации.

На каждом этапе проектирования информационной модели будущей системы разрабатывается соответствующая модель данных: логическая и физическая модели [15].

### **Логическое моделирование**

Перед проектированием информационной системы необходимо приступить к созданию логической модели по методологии IDEF1X с помощью Microsoft Visio [5]. Логическая модель представляет собой модель базы данных, которая не привязана к контрактной СУБД [15].

Основными объектами модели является сущность связь, ключ, атрибут. Так же связь имеет степень, то есть количество типов сущностей, которые участвуют в данной связи. Возможны варианты связи: бинарная– это связь между двумя разными сущностями. Тенарная – это связь между 3 сущностями. При моделировании использовалась бинарные связи так как необходимо связывать две разные сущности [25].

ERD-диаграмма составлена в трёх уровнях: Уровень «сущность-связь» (ER), Уровень ключей (KB), Уровень атрибутов (FA).

Логическая модель FA-уровня спроектирована с использованием IDEF1X методологии (см. Приложение 4).

Всякое отношение имеет некоторую интерпретацию, передающую смысл представленных в нем данных. Эта интерпретация может быть сформулирована в виде предиката. Предикаты отношений в неформальной записи предметной области:

— CHECK\_VACCINE: пользователь с определенным идентификатором (ID\_USER) проверяется на вакцинацию

(ID\_CHECK\_VACCINE) в определенную дату (Date), с температурой тела (BodyTemp), результатом осмотра (ResultInspeccion), привит вакциной (Vaccine), имеет разрешение (Permission), и имеет состояние через тридцать минут (StateAfterThirtyMin).";

— DATA\_MEDICAL\_CHECKUPS: пользователь с определенным идентификатором (ID\_USER) проходит медицинский осмотр (ID\_DATA\_MEDICAL\_CHECKUPS) в определенном году (Year) в медицинской организации (ID\_MEDICAL\_ORGANIZATION), при этом осмотр содержит информацию о виде обследования (Survey) и его результате (SurveyResult);

— GROUP: группа с определенным идентификатором (ID\_GROUP) имеет название (GroupName) и количество членов (NumberPeople);

— HOSPITAL\_TREATMENT: пользователь с определенным идентификатором (ID\_USER) проходит лечение в больнице (ID\_HOSPITAL\_TREATMENT) с описанием (Description);

— LOGIN\_PASSWORD: пользователь с определенным идентификатором (ID\_USER) имеет логин (Login) и пароль (Password).";

— MEDICAL\_ORGANIZATION: медицинская организация с определенным идентификатором (ID\_MEDICAL\_ORGANIZATION) имеет название (Name) и адрес (Address);

— OPERATIONS: пользователь с определенным идентификатором (ID\_USER) проходит операцию (ID\_OPERATIONS) с описанием (Operations);

— PASS\_CONROLLER: контролер пропусков с определенным идентификатором (ID\_PASS\_CONROLLER) связан с пользователем (ID\_USER);

— PASSES\_FOR\_ILLNESS: пользователь с определенным идентификатором (ID\_USER) имеет пропуск по болезни (ID\_PASSES\_FOR\_ILLNESS) с определенной датой начала (DateFrom), датой окончания (DateTo) и диагнозом (Diagnosis)

— PAST\_INFECTIONOUS\_DIASES: пользователь с определенным идентификатором (ID\_USER) имеет прошедшие инфекционные болезни (ID\_PAST\_INFECTIONOUS\_DIASES) с датой (Date) и описанием (Description);

— PREVENTIVE\_VACCINE: пользователь с определенным идентификатором (ID\_USER) получает профилактическую вакцину (ID\_PREVENTIVE\_VACCINE) в определенной медицинской организации (ID\_MEDICAL\_ORGANIZATION), вакцина имеет название (Vaccine) и дату (Date);

— SANITARY\_SPA\_TREATMENT: пользователь с определенным идентификатором (ID\_USER) проходит санаторно-курортное лечение (ID\_SANITARY\_SPA\_TREATMENT) с описанием (Description);

— SURVEYS: пользователь с определенным идентификатором (ID\_USER) проходит обследование (ID\_SURVEYS) в медицинской организации (ID\_MEDICAL\_ORGANIZATION) в определенном году (Year), обследование содержит информацию о виде обследования (Survey) и его результате (SurveyResult);

— TRAUMA: пользователь с определенным идентификатором (ID\_USER) получает травму (ID\_TRAUMA) с описанием (Description);

— USER: пользователь с определенным идентификатором (ID\_USER) имеет роль (ID\_USER\_ROLE), фамилию (Surname), имя (Name), отчество (Patronymic), дату рождения (DateBirth), домашний адрес (HomeAddress), проживает в общежитии (Hostel), является членом группы (ID\_GROUP), дату зачисления (DateEnrollment), цвет (Color), и название изображения (ImageName);

— USER\_ROLE: роль пользователя с определенным идентификатором (ID\_USER\_ROLE) имеет название (NameRole).

## **Составление глоссариев и спецификаций**

Глоссарий – это словарь, в котором содержатся толкования всех слов, которые могут быть незнакомы читателю. Глоссарий значительно облегчает восприятие текста, так как человек в любой момент имеет возможность обратиться к словарю и проверить значение определённого термина.

После выявления предикатов отношений был сформирован глоссарий сущностей, содержащий наименование сущности и её описание. Глоссарии могут формироваться разными способами: при помощи мастера отчётов в программах для проектирования и документирования баз данных или вручную в текстовых редакторах. Исходя из логической модели были сформированы глоссарий сущностей (см. Приложение 1).

После составления глоссария сущностей был сформирован глоссарий атрибутов, содержащий наименование атрибута, наименование сущности атрибута, тип и описание атрибута (см. Приложение 2).

## **Создание физической модели базы данных**

Прежде чем приступать к созданию физической модели в системе управления базами данных (СУБД), необходимо тщательно выбрать одну из нескольких доступных СУБД для дальнейшей реализации проекта.

Базы данных – это хранилища, в которых содержатся большие массивы информации. Отправлять запросы, получать информацию из них можно при помощи СУБД – систем управления [39]. СУБД служит посредником между пользователем и БД. Сама структура БД хранится в виде набора файлов, и единственный способ получить доступ к данным в этих файлах – через СУБД [10].

Каждая СУБД должна удовлетворять следующим требованиям:

- надежность. СУБД должна обеспечивать надежное хранение данных и предотвращать их потерю или повреждение;
- безопасность. СУБД должна обеспечивать безопасность данных, предоставляя механизмы аутентификации и авторизации, а также шифрование и контроль доступа;
- масштабируемость. СУБД должна быть способна масштабироваться в соответствии с растущими потребностями в хранении и обработке данных;
- производительность. СУБД должна обеспечивать высокую производительность при обработке запросов и транзакций;

При сравнении возможностей, достоинств и недостатков разных систем управления базами данных выбор пал на Microsoft SQL Server. Выбор Microsoft SQL Server обусловлен рядом стратегических и технических факторов, которые существенно влияют на обработку и хранение медицинских данных в контексте медицинского кабинета колледжа.

1. Безопасность. SQL Server предоставляет мощные механизмы безопасности и соответствует стандартам соблюдения законодательства в области медицинской конфиденциальности, что является приоритетом при обработке медицинских данных;

2. Производительность и Масштабируемость. Высокая производительность SQL Server, особенно в контексте транзакционных операций и запросов, обеспечивает эффективную обработку и доступ к данным, что крайне важно для оперативной работы медицинского кабинета;

3. Гибкость и Надежность. Система предоставляет гибкие инструменты для разработки и масштабируемость, позволяя легко адаптироваться к изменяющимся требованиям и обеспечивая надежность в хранении ценных медицинских данных;

Физическая модель автоматизированной информационной системы «Контроль состояния здоровья студентов ГАПОУ СО «Камышловский

педагогический колледж» была создана путем переноса логической модели FA уровня в SQL Server Management (см. Приложение 5).

После создания физической модели базы данных выявлены следующие схемы реляционных таблиц:

- USER (ID\_USER, ID\_USER\_ROLE, Surname, Name, Patronymic, DateBirth, HomeAddress, Hostel, ID\_GROUP, DateEnrollment, Color, ImageName);
- GROUP (ID\_GROUP, GroupName, NumberPeople);
- CHECK\_VACCINE (ID\_CHECK\_VACCINE, ID\_USER, Date, BodyTemp, ResultInspeccion, Vaccine, Permission, StateAfterThirtyMin);
- DATA\_MEDICAL\_CHECKUPS (ID\_DATA\_MEDICAL\_CHECKUPS, ID\_USER, ID\_MEDICAL\_ORGANIZATION, Year, Survey, SurveyResult);
- HOSPITAL\_TREATMENT (ID\_HOSPITAL\_TREATMENT, ID\_USER, Description);
- LOGIN\_PASSWORD (Login, Password, ID\_USER);
- MEDICAL\_ORGANIZATION (ID\_MEDICAL\_ORGANIZATION, Name, Address);
- OPERATIONS (ID\_OPERATIONS, ID\_USER, Operations);
- PASS\_CONROLLER (ID\_PASS\_CONROLLER, ID\_USER);
- PASSES\_FOR\_ILLNESS (ID\_PASSES\_FOR\_ILLNESS, ID\_USER, DateFrom, DateTo, Diagnosis);
- PAST\_INFECTIOUS\_DIASES (ID\_PAST\_INFECTIOUS\_DIASES, ID\_USER, Date, Description);
- PREVENTIVE\_VACCINE (ID\_PREVENTIVE\_VACCINE, ID\_USER, ID\_MEDICAL\_ORGANIZATION, Vaccine, Date);
- SANITARY\_SPA\_TREATMENT (ID\_SANITARY\_SPA\_TREATMENT, ID\_USER, Description);

- SURVEYS (ID\_SURVEYS, ID\_USER, Year, Survey, SurveyResult, ID\_MEDICAL\_ORGANIZATION);
- TRAUMA (ID\_TRAUMA, ID\_USER, Description);
- USER\_ROLE (ID\_USER\_ROLE, NameRole).

## Использование импорта

На этом этапе проектирования физической модели базы данных исходные файлы готовятся для импорта (см. Рисунок 8).

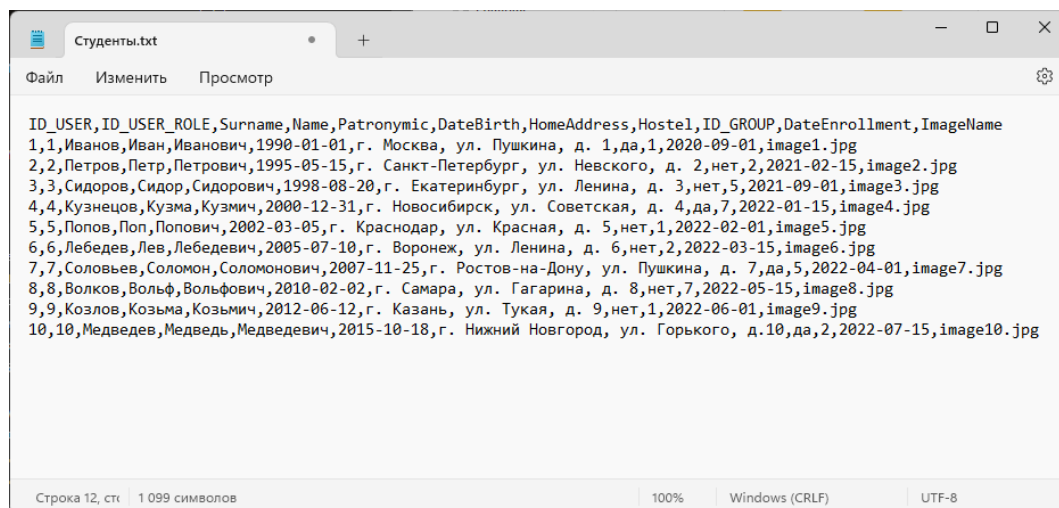


Рисунок 8 - Данные для импорта

Этот процесс может включать форматирование неструктурированных файлов для использования их в импорте в базу данных. Были созданы файлы для импорта данных с медицинскими данными студентов.

После того, как исходные файлы были подготовлены для импорта, выполняется процесс импорта данных в базу данных. Для этого создается временная база данных, в которую импортируются данные для последующего переноса в основную базу данных. С помощью мастера импорта и экспорта SQL Server производится импорт файлов формата \*.csv, \*.xls и \*.txt в временную базу данных (см. Рисунок 9).



PC\SQLXPRESS.DiplomOboskalov - dbo.import\_студенты - Microsoft SQL Server Management Studio

ID_USER	ID_USER_ROLE	Surname	Name	Patronymic	DateBirth	HomeAddress	Hostel	ID_GROU
1	1	Иванов	Иван	Иванович	1990-01-01	г. Москва, ул. ...	да	1
2	2	Петров	Петр	Петрович	1995-05-15	г. Санкт-Петер...	нет	2
3	3	Сидоров	Сидор	Сидорович	1998-08-20	г. Екатеринбург...	нет	5
4	4	Кузнецов	Кузма	Кузмич	2000-12-31	г. Новосибирс...	да	7
5	5	Попов	Поп	Попович	2002-03-05	г. Краснодар, у...	нет	1
6	6	Лебедев	Лев	Лебедевич	2005-07-10	г. Воронеж, ул...	нет	2
7	7	Соловьев	Соломон	Соломонович	2007-11-25	г. Ростов-на-Д...	да	5
8	8	Волков	Вольф	Вольфович	2010-02-02	г. Самара, ул. ...	нет	7
9	9	Козлов	Козьма	Козьмич	2012-06-12	г. Казань, ул. Т...	нет	1
10	10	Медведев	Медведь	Медведевич	2015-10-18	г. Нижний Нов...	да	2
NULL	NULL	NULL	NULL	NULL	NULL	NULL	NULL	NULL

Рисунок 9 - Временная таблица для импортированных данных

Затем, используя язык структурирования запросов SQL, выполняется копирование данных в основную базу данных (см. Рисунок 10).

SQLQuery4.sql - PC\SQLXPRESS.DiplomOboskalov (PC\savevy (58))\* - Microsoft SQL Server Management Studio

```

INSERT INTO [USER] (ID_USER_ROLE, Surname, [Name], Patronymic, DateBirth,
HomeAddress, Hostel, ID_GROUP, DateEnrollment, Color, ImageName)
SELECT [ID USER_ROLE]
, [Surname]
, [Name]
, [Patronymic]
, [DateBirth]
, [HomeAddress]
, [Hostel]
, [ID GROUP]
, [DateEnrollment]
, [Color]
, [ImageName]
FROM [DiplomOboskalov].[dbo].[import_студенты]

```

Сообщения

(затронута строк: 10)

Время выполнения: 2024-06-19T22:15:09.8151691+05:00

Запрос успешно выполнен.

Рисунок 10 - SQL запрос на вставку данных

## Создание SQL скрипта базы данных

После того как база данных была разработана в SQL Server, был создан полный скрипт базы данных, включающий все таблицы и данные (см. Рисунок 11). Этот скрипт, созданный на основе базы данных, представляет собой набор инструкций SQL, которые могут быть использованы для создания, изменения или восстановления базы данных. Он может включать команды для создания таблиц, индексов, ограничений и триггеров, а также для вставки, обновления или удаления данных.

Этот скрипт может быть использован для создания копии базы данных на тестовом или продуктивном сервере, для выполнения регулярных обновлений или исправлений, а также для восстановления базы данных в случае ее повреждения или потери

```
USE [OboskalovDiplom]
GO
/***** Object: Table [dbo].[USER]    Script Date: 22.02.2024 13:40:36 *****/
SET ANSI_NULLS ON
GO
SET QUOTED_IDENTIFIER ON
GO
CREATE TABLE [dbo].[USER](
    [ID_USER] [int] IDENTITY(1,1) NOT NULL,
    [ID_USER_ROLE] [int] NOT NULL,
    [Surname] [nvarchar](50) NOT NULL,
    [Name] [nvarchar](50) NOT NULL,
    [Patronymic] [nvarchar](50) NULL,
    [DateBirth] [date] NOT NULL,
    [HomeAddress] [nvarchar](50) NULL,
    [Hostel] [char](5) NULL,
    [ID_GROUP] [int] NULL,
    [DateEnrollment] [date] NULL,
    [Color] [nvarchar](50) NULL,
    [ImageName] [nvarchar](50) NULL,
    CONSTRAINT [PK_User] PRIMARY KEY CLUSTERED
(
    [ID_USER] ASC
)WITH (PAD_INDEX = OFF, STATISTICS_NORECOMPUTE = OFF,
    IGNORE_DUP_KEY = OFF, ALLOW_ROW_LOCKS = ON,
    ALLOW_PAGE_LOCKS = ON, OPTIMIZE_FOR_SEQUENTIAL_KEY = OFF) ON [PRIMARY]
) ON [PRIMARY]
GO
/***** Object: Table [dbo].[GROUP]    Script Date: 22.02.2024 13:40:36 *****/
SET ANSI_NULLS ON
GO
SET QUOTED_IDENTIFIER ON
```

Рисунок 11 - Фрагмент SQL скрипта базы данных

## **Проектирование приложений конечных пользователей**

### **Выбор среды разработки приложений для ИС**

Из множества существующих языков программирования я выбрал C#, поскольку он является одним из самых популярных и широко используемых языков программирования во всем мире. Этот язык предоставляет богатую библиотеку классов и инструментов, поддерживает объектно-ориентированное программирование и хорошо интегрируется с системами управления базами данных. Для программирования на C# необходима совместимая с операционной системой интегрированная среда разработки.

Интегрированная среда разработки (Integrated Development Environment, IDE) представляет собой комплексное программное обеспечение, которое объединяет в себе необходимые инструменты для разработки программного обеспечения. Она включает в себя текстовый редактор, компилятор или интерпретатор, средства автоматизации сборки и отладчик.

Для выбора наиболее подходящей среды разработки для программирования на языке C# было проведено сравнение нескольких популярных интегрированных сред разработки, таких как Microsoft Visual Studio, JetBrains Rider, Visual Studio Code и Sharpdevelop. Были рассмотрены преимущества и недостатки каждой из них, чтобы сделать обоснованное решение о выборе наиболее подходящей среды разработки.

1. Microsoft Visual Studio является одной из самых популярных и широко используемых интегрированных сред разработки для программирования на языке C#. Она предоставляет широкий спектр возможностей для разработки, отладки и тестирования приложений, а также имеет удобный и настраиваемый интерфейс. Кроме того, Visual Studio поддерживает работу с различными системами управления версиями и предоставляет множество полезных инструментов для разработки.

2. JetBrains Rider является относительно новым продуктом компании JetBrains, который также предназначен для программирования на языке C#. Эта интегрированная среда разработки имеет множество преимуществ, таких как высокая производительность, настраиваемый интерфейс, поддержка работы с различными системами управления версиями и множество полезных инструментов для разработки.

3. Visual Studio Code является легковесной и гибкой интегрированной средой разработки, которая также поддерживает программирование на языке C#. Она предоставляет множество полезных расширений для разработки, отладки и тестирования приложений, а также имеет удобный и настраиваемый интерфейс.

4. Sharpdevelop является бесплатной и открытой интегрированной средой разработки для программирования на языке C#. Она предоставляет множество полезных инструментов для разработки, отладки и тестирования приложений, а также имеет удобный и настраиваемый интерфейс.

В итоге, после сравнения преимуществ и недостатков каждой из интегрированных сред разработки, было принято решение выбрать Microsoft Visual Studio в качестве основной среды разработки для программирования на языке C#. Это решение было обосновано широким спектром возможностей, предоставляемых Visual Studio, ее высокой производительностью и надежностью, а также наличием множества полезных инструментов для разработки, отладки и тестирования приложений.

## **Проектирование с использованием UML диаграмм**

UML — это унифицированный язык моделирования. Его используют, чтобы создавать диаграммы и схемы для визуализации процессов и явлений [16]. Диаграммы UML подразделяют на два типа — это структурные диаграммы и диаграммы поведения. По анализу предметной области

определен тип диаграммы поведения. Диаграммы поведения отображают динамическое поведение объектов в системе, которое можно описать, как серию изменений в системе с течением времени [11].

Диаграмма деятельности.

Диаграмма деятельности UML обеспечивает визуальное представление бизнес-процессов, поддерживая моделирование последовательностей, условий, параллельных операций, циклов и потоков данных (см. Приложение 3). Она улучшает понимание бизнес-логики, решений и взаимодействия между компонентами системы, делая её полезной для анализа и документирования бизнес-процессов.

Диаграмма представляет взаимодействие между Информационной системой и Базой данных при создании медицинского осмотра для студента. Информационная система начинает с запроса на создание медицинского осмотра и поиска студента в базе данных. Если студент не найден, система сообщает об этом и возвращает к запросу на поиск. В случае наличия студента, система создает форму для заполнения медицинского осмотра, после чего происходит заполнение данных и запрос на сохранение записи в базе данных. База данных проверяет соответствие данных формату и ограничениям. Если данные не соответствуют, система сообщает об ошибке и возвращает к заполнению формы. В случае успешного сохранения, система завершает процесс, сообщая об успешном сохранении данных.

Диаграмма последовательности.

Диаграмма последовательности UML предоставляет визуальное представление последовательности взаимодействия между объектами в системе (см. Рисунок 12). Она позволяет визуализировать и описывать шаги выполнения сценария, отслеживать временные аспекты, выделять параллельные потоки выполнения, моделировать альтернативные пути и анализировать использование системы, что делает её полезной для документирования процессов и обнаружения потенциальных проблем в системе.

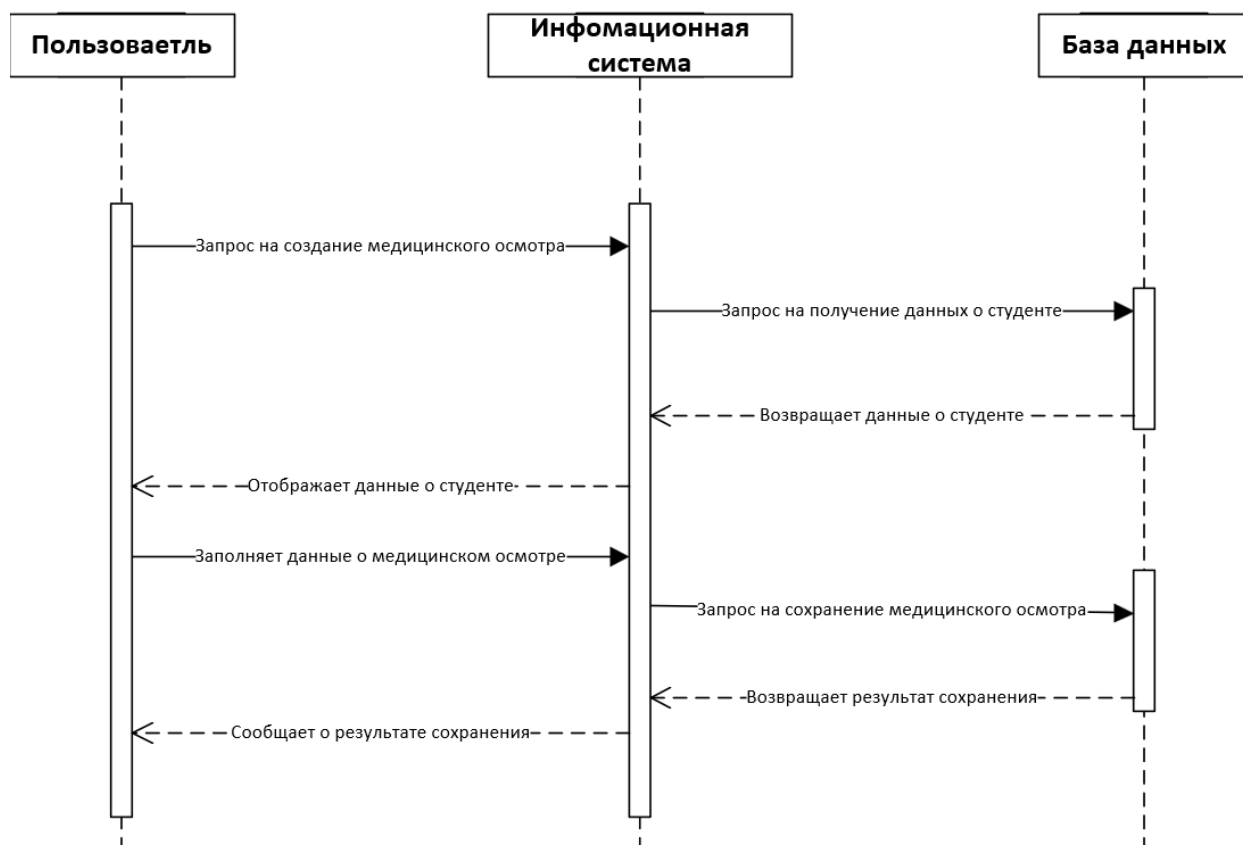


Рисунок 12 - Диаграмма последовательности «Создание записи Медицинский осмотр»

Диаграмма последовательности UML описывает взаимодействие между тремя сущностями: Пользователь, Информационная система и База данных в контексте запроса на создание медицинского осмотра. Диаграмма иллюстрирует взаимодействие между пользователями, информационной системой и базой данных в процессе создания медицинского осмотра.

Описание последовательности событий: последовательность событий начинается с инициации создания медицинского осмотра пользователем. Затем информационная система получает запрос и запрашивает необходимые данные о студенте у базы данных. База данных возвращает данные, которые отображаются пользователю. Пользователь заполняет данные о медицинском осмотре и инициирует сохранение. Информационная система направляет запрос на сохранение в базу данных, которая сохраняет данные и возвращает результаты. Информационная система сообщает пользователю о результатах

сохранения. В конце пользователь завершает свой Жизненный Цикл после получения информации о результате сохранения.

Диаграмма состояний.

Диаграмма состояний UML моделирует различные состояния и переходы между ними в процессе взаимодействия пользователя с приложением для формирования отчёта по медицинским документам студента (см. Рисунок 13).



Рисунок 13 - Диаграмма состояний «Создание записи Медицинский осмотр»

Описание каждого состояния и перехода: процесс начинается с исходного состояния "Начальное состояние", затем переходит в "Вход в приложение", где пользователь вводит свои учетные данные. После успешного входа процесс переходит в "Формирование отчёта по медицинским документам студента", где приложение начинает формировать отчёт на основе медицинских данных студента. Затем процесс переходит в "Заполнение отчёта", где пользователь вводит необходимые данные для отчёта. После заполнения данных процесс переходит в "Проверка введённых данных", где система проверяет данные на корректность. Если данные проходят проверку, процесс переходит в "Сохранение отчёта", где отчёт сохраняется в базе данных. После успешного сохранения процесс переходит в "Печать отчёта", где пользователь может выбрать опцию печати отчёта. После печати отчёта процесс переходит в "Отчёт распечатан", где приложение подтверждает

успешную печать отчёта. В конце процесс переходит в "Конец", представляющее завершение процесса.

Эта диаграмма состояний позволяет визуализировать последовательность действий пользователя в приложении для формирования отчёта по медицинским документам студента и ясно представляет переходы между различными состояниями в процессе выполнения задачи.

## **Программная реализация информационной системы**

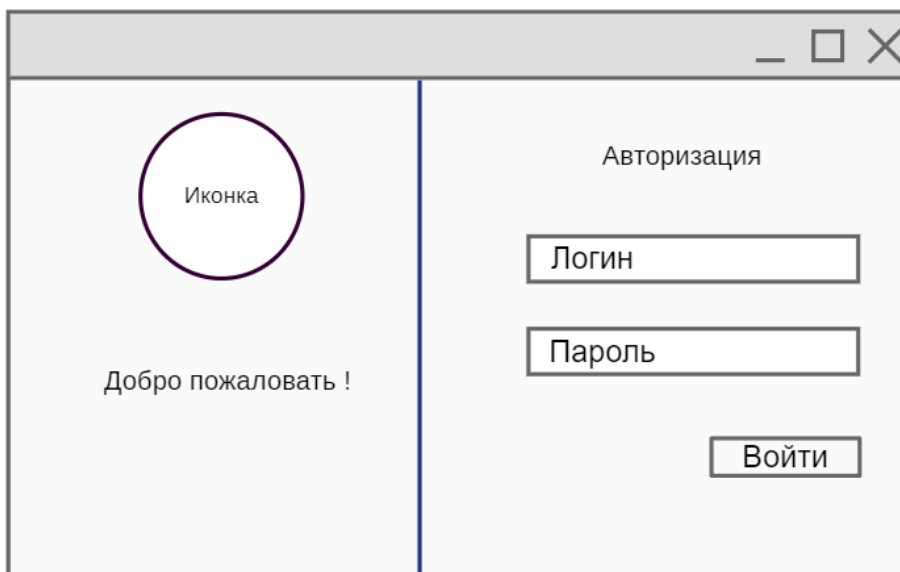
### **Разработка интерфейса приложений**

Программные средства реализации информационных процессов – это специальные программы и инструменты, которые используются для создания, разработки и выполнения информационных процессов в компьютерных системах [17].

При создании приложения разрабатываются детальные макеты основных форм и функций, чтобы обеспечить удобство использования и эффективную работу пользователей. Каждая деталь была продумана до мелочей, чтобы обеспечить плавный и интуитивно понятный пользовательский опыт [29]. Целью было разработать приложение, которое будет не только функциональным, но и привлекательным визуально, чтобы привлечь и



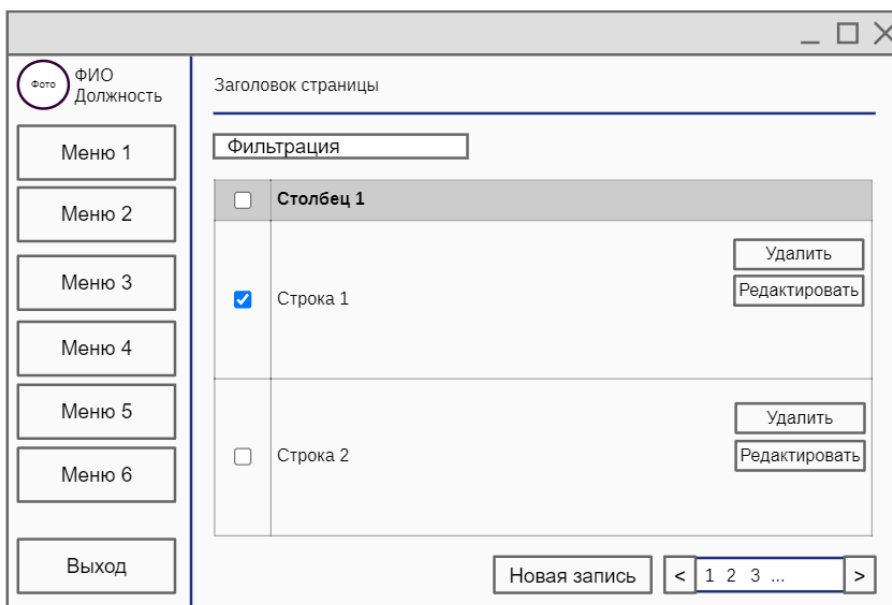
удержать пользователей [38]. Разрабатываемый макет для входа в программу (см. Рисунок 14).



The mockup shows a window titled 'Авторизация' (Authorization). It is divided into two sections by a vertical line. The left section contains a circular placeholder labeled 'Иконка' (Icon) and the text 'Добро пожаловать !' (Welcome!). The right section contains two input fields labeled 'Логин' (Login) and 'Пароль' (Password), and a 'Войти' (Login) button.

Рисунок 14 - Макет окна авторизации

Основные функции и информация представлена на главной форме приложения, где в дальнейшем будут осуществляются вывод информации (см. Рисунок 15).



The mockup shows a window with a sidebar on the left and a main content area. The sidebar contains a user profile section with 'Фото' (Photo), 'ФИО' (Full Name), and 'Должность' (Position), followed by buttons 'Меню 1' through 'Меню 6' and a 'Выход' (Exit) button. The main content area has a 'Заголовок страницы' (Page Header) section with a 'Фильтрация' (Filter) input field. Below this is a table with one column labeled 'Столбец 1' (Column 1). The table contains two rows: 'Строка 1' (Row 1) and 'Строка 2' (Row 2). Each row has a checkbox and buttons 'Удалить' (Delete) and 'Редактировать' (Edit). At the bottom, there is a 'Новая запись' (New Record) button and a pagination control showing '1 2 3 ...'.

Рисунок 15 - Макет главного окна приложения

В приложении доступны функции добавление и редактирование осуществляться функции будет с одной формы (см. Рисунок 16).

Добавление записи

Фото

Данные 1

Данные 2

Данные 3

Данные 4

Данные 5

Данные 6

Сохранить

Рисунок 16 - Страница добавления записи

### Структура интерфейса

Главное меню. Меню, расположенное в левой части интерфейса, содержит основные категории и функции системы. Здесь пользователь может найти доступ к основным разделам и действиям (см. Рисунок 17).

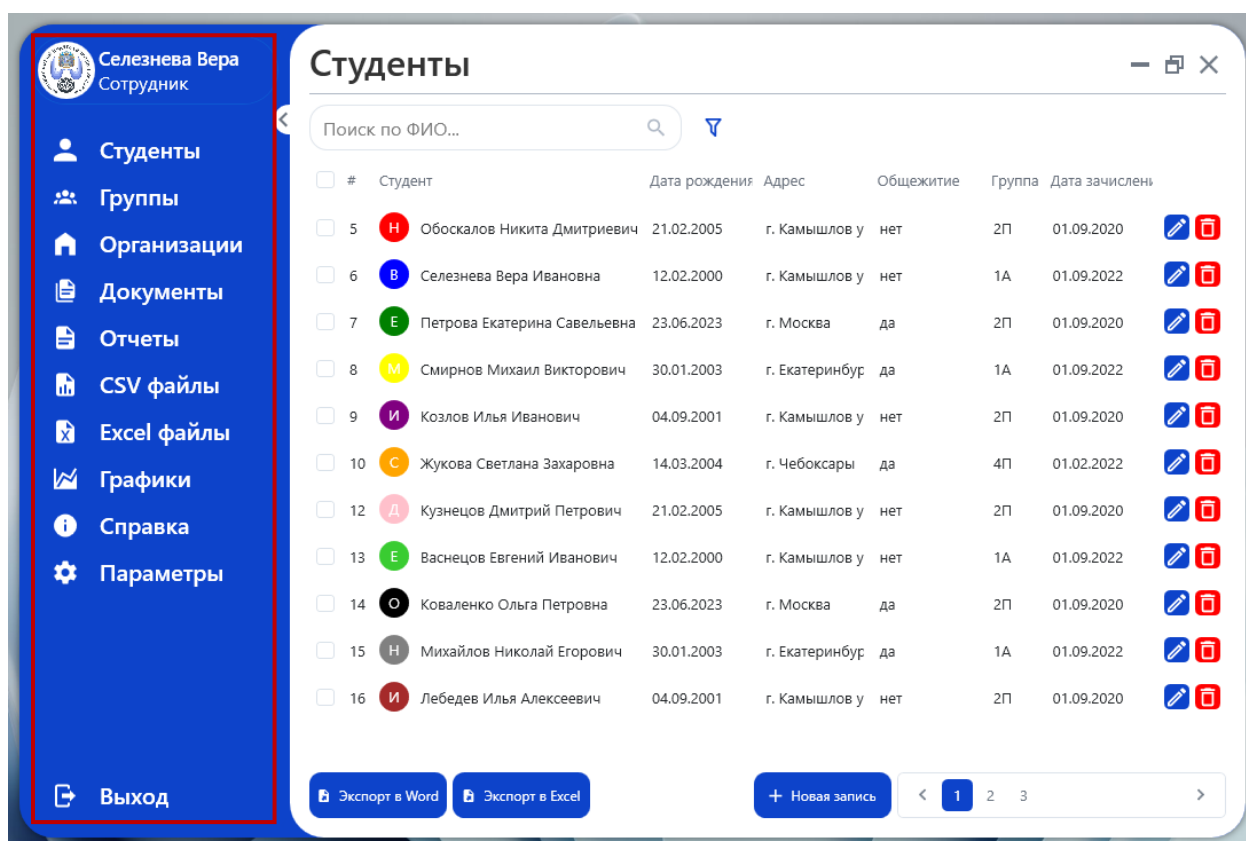


Рисунок 17 - Меню приложения

Область рабочего пространства. Основное окно, где отображается основная информация и контент. В этой области пользователь взаимодействует с данными, выполняет задачи и просматривает результаты операций (см. Рисунок 18).

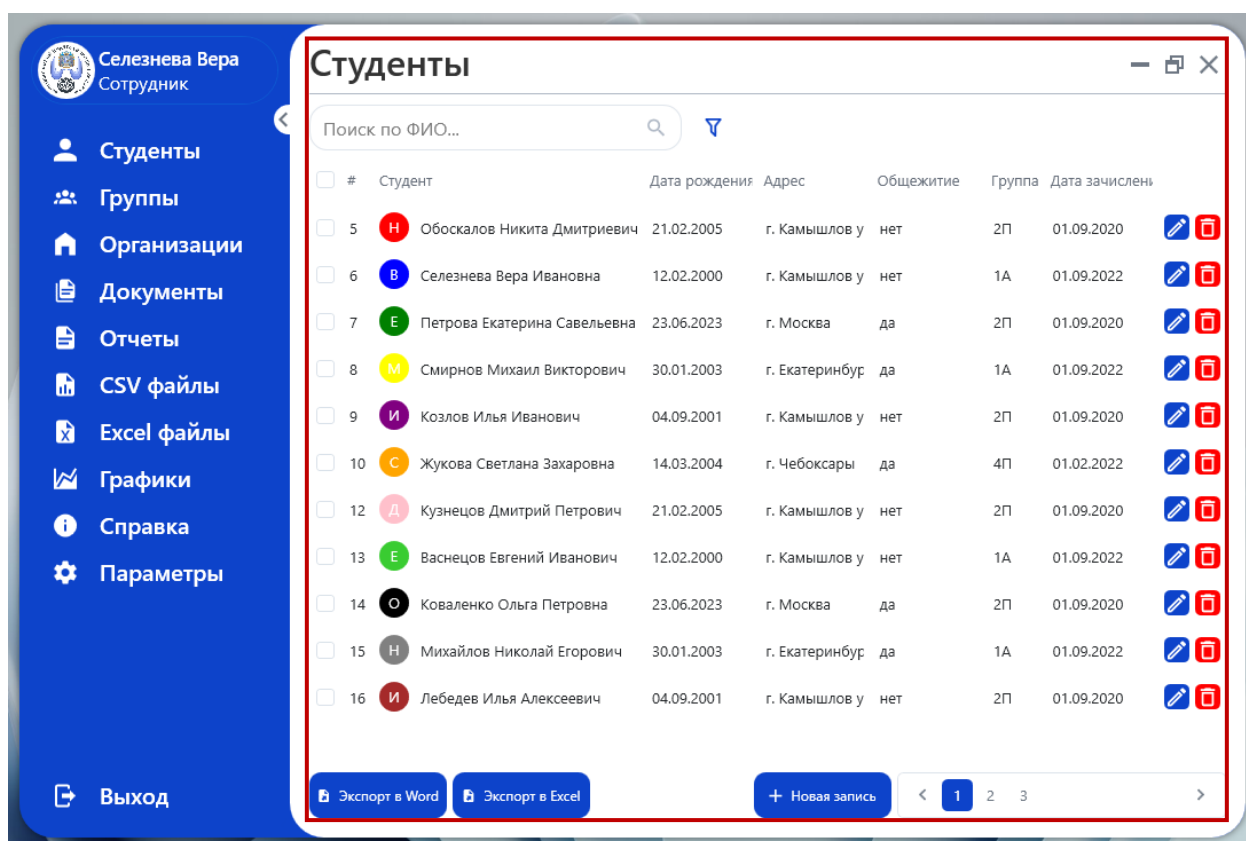


Рисунок 18 - Область рабочего пространства

Формы и диалоговые окна. Элементы, предоставляющие пользователю возможность ввода данных или выбора опций. Формы могут использоваться для создания, редактирования или поиска информации (см. Рисунок 19).

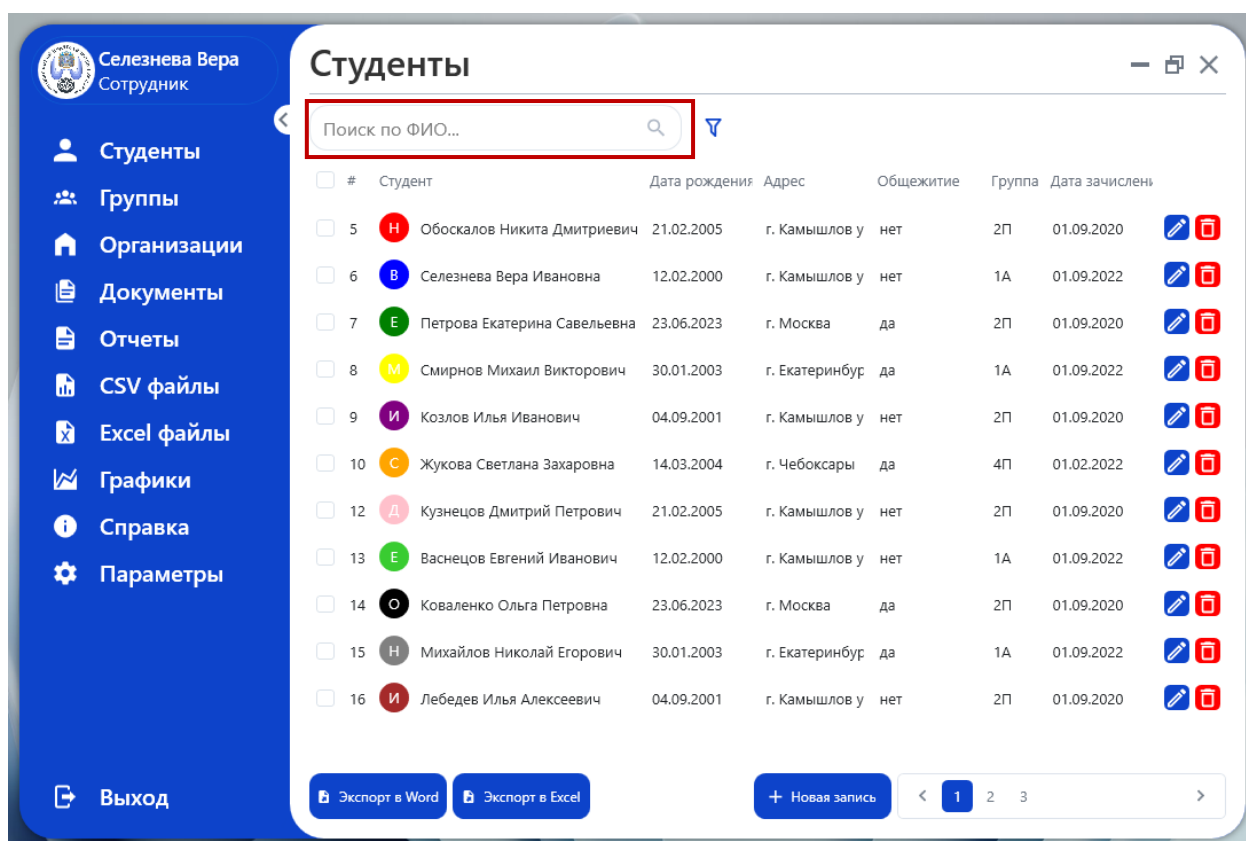


Рисунок 19 – Элементы ввода данных

Элементы управления. Кнопки, чекбоксы, радиокнопки, выпадающие списки и другие элементы, предоставляющие пользователю средства взаимодействия с системой (см. Рисунок 20).

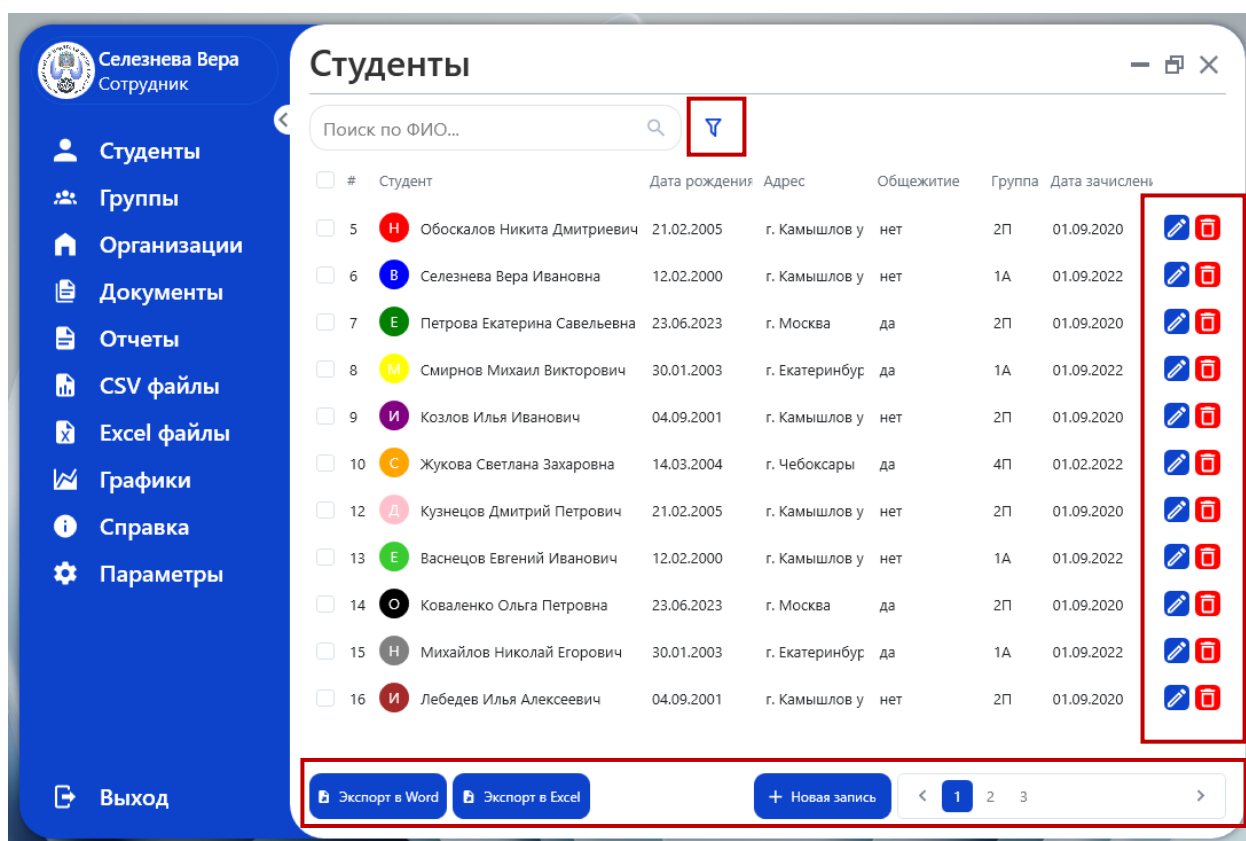


Рисунок 20 - Элементы управления

### Описание меню

Меню в программном интерфейсе представляет собой структурированный набор команд, опции и функций, предоставляемых пользователю для управления программой или системой. Основные компоненты и характеристики меню:

1. Студенты. Элемент меню "Студенты" открывает страницу со списком студентов из базы данных с постраничным просмотром, кнопками для удаления, редактирования и добавления студента, а также фильтрами для настройки отображения данных по заданным критериям;

2. Группы. Элемент меню "Группы" открывает страницу со списком всех групп и студентов в выбранной группе из базы данных. Кнопки редактирования, добавления и удаления групп облегчают администрирование, а фильтры позволяют настраивать отображение данных по выбранным параметрам;

3. Организации. На странице "Медицинские организации"

отображаются все организации из базы данных. Пользователь может управлять данными с помощью кнопок удаления, редактирования и добавления организаций, а также использовать фильтры для персонализированного просмотра данных;

4. Документы. Страница документов предоставляет пользователю удобный доступ ко всем документам, доступным для него. Информация о документах представлена в удобной форме;

5. Отчеты. Страница отчетов предоставляет пользователю перечень доступных отчетов. Пользователь может просматривать и загружать отчеты, необходимые ему для работы;

6. CSV файлы. На странице выбора CSV файлов пользователь может загружать CSV файлы с компьютера для вывода данных. Это предоставляет удобный способ импорта информации из внешних источников;

7. Excel файлы. Страница выбора Excel файлов позволяет пользователю выбирать и загружать Excel файлы с компьютера для дополнительного анализа данных;

8. Графики. Элемент меню "Графики" дает возможность просмотра динамического списка графиков, обеспечивая визуализацию данных;

9. Справка. Страница со списком файлов, полезных для пользователя, предоставляет доступ к справочной информации, руководствам и другим документам, помогающим в использовании приложения;

10. Параметры. Страница настроек приложения предоставляет пользователю возможность выбора и изменения параметров приложения, что позволяет настраивать интерфейс и функциональность под свои предпочтения;

11. Выход. Элемент меню "Выход" открывает окно авторизации пользователя, позволяя ему безопасно завершить сеанс работы в приложении.

## Многопользовательский режим

Работа многопользовательского режима в приложении реализована с помощью дополнительного окна при запуске. Для входа необходимо правильно ввести пароль и логин (см. Рисунок 21).

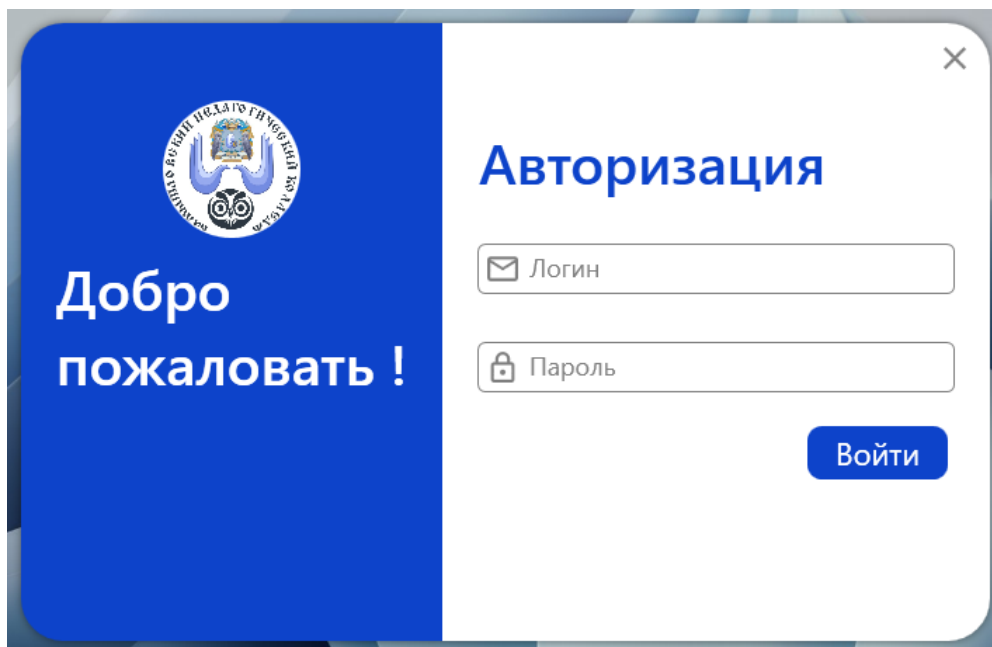


Рисунок 21 - Окно авторизации

Если логин или пароль были неверно введены более 3 раз, то ввод данных будет заблокирован и пользователю будет необходимо ввести код капчи (см. Рисунок 22)



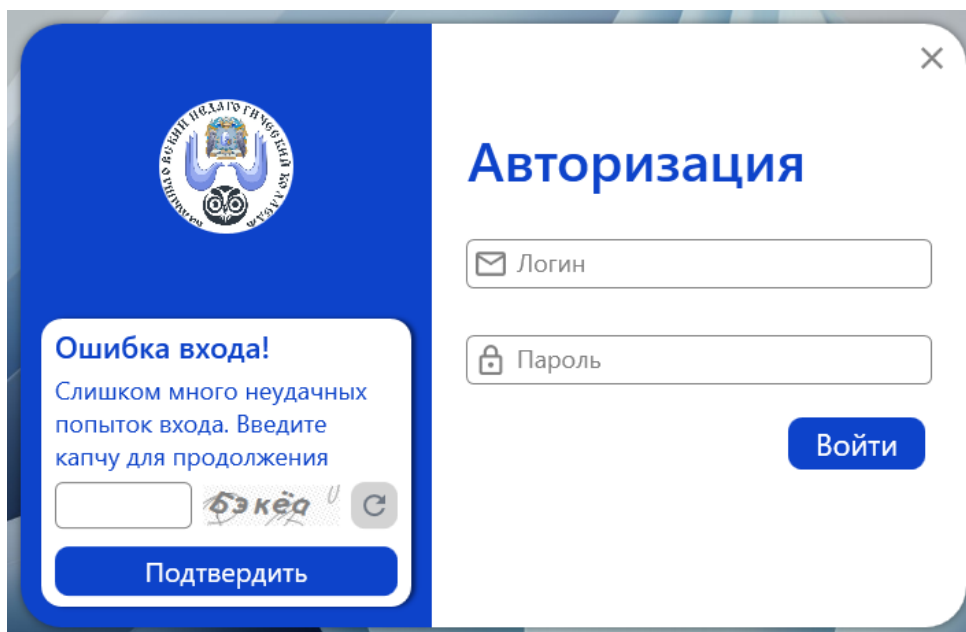


Рисунок 22 - Вывод каптчи

При не заполнении полей логина или пароля появятся окна с ошибками (см. Рисунок 23)

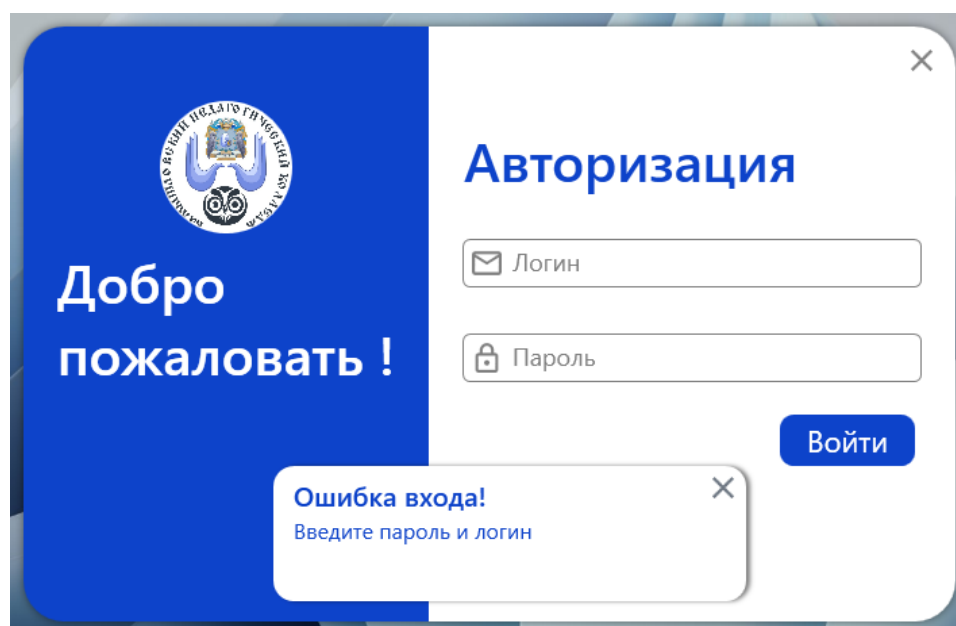


Рисунок 23 - Ошибка при не заполнении полей

Если пользователь введет несуществующие логин или пароль, система оповестит об этом (см. Рисунок 24).

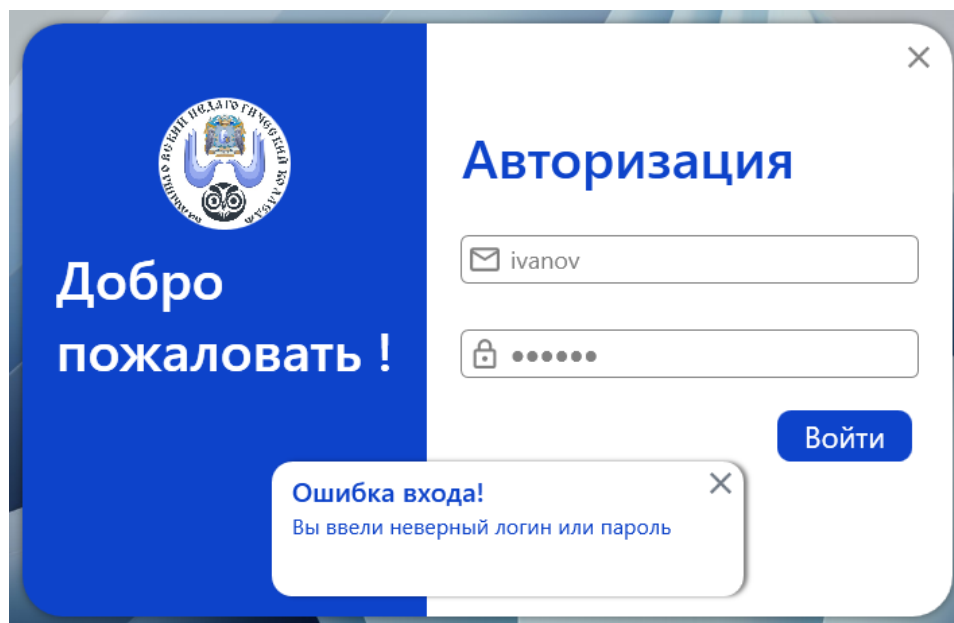


Рисунок 24 - Ошибка при вводе неправильных данных

При правильном вводе данных будет открыта домашняя страница приложения.

### **Реализация функций добавление, удаление, редактирование**

В приложении реализованы функции добавление, удаление, редактирование записей при помощи дополнительных форм для заполнения данных. Данные функции предоставлены каждому авторизованному пользователю [22].

Чтобы создать новую запись в любой форме приложения, необходимо нажать кнопку "Новая запись" и ввести все необходимые данные в открывшемся окне. После заполнения всех полей нажмите кнопку "Сохранить", чтобы сохранить новую запись. Например, при создании записи о студенте (см. Рисунок 25), можно добавить данные о студенте и загрузить фото студента.

Селезнева Вера  
Сотрудник

← **Добавление студента**

Фамилия: Обоскалов

Имя: Никита

Отчество: Дмитриевич

Дата рождения: 21.02.2005

Адрес: г. Камышлов

Общежитие: нет

Группа: 4П

Дата зачисления: 01.09.2020

Изменить

Сохранить

Выход

Рисунок 25 - Окно добавления записи

При успешном добавлении записи появится сообщение (см. Рисунок 26).

Селезнева Вера  
Сотрудник

← **Добавление студента**

Фамилия: Обоскалов

Имя: Никита

Отчество: Дмитриевич

Дата рождения: 21.02.2005

Адрес: г. Камышлов

Общежитие: нет

Группа: 4П

Дата зачисления: 01.09.2020

Изменить

Сохранить

Выход

**Сообщение**

Запись успешно добавлена

Рисунок 26 - Сообщение при успешном добавлении данных

В этом же окне реализовано изменение записей. Пользователь может открыть окно изменения по кнопке из основного окна или по пункту контекстного меню. Тогда окно изменения откроется с загруженными из базы данных полями выбранной строки (см. Рисунок 27).

The screenshot shows a web application interface. On the left is a blue sidebar with a user profile at the top: 'Селезнева Вера' (Selyznova Vera), 'Сотрудник' (Employee). Below are menu items: 'Студенты' (Students), 'Группы' (Groups), 'Организации' (Organizations), 'Документы' (Documents), 'Отчеты' (Reports), 'CSV файлы' (CSV files), 'Excel файлы' (Excel files), 'Графики' (Charts), 'Справка' (Help), 'Параметры' (Parameters), and 'Выход' (Exit) at the bottom. The main window is titled '← Редактирование записи о студенте' (← Editing student record). It features a student icon (a person in a green graduation cap and gown) with a blue 'Изменить' (Edit) button below it. To the right of the icon are form fields for: 'Фамилия:' (Oboskalov), 'Имя:' (Nikita), 'Отчество:' (Dmitrievich), 'Дата рождения:' (21.02.2005), 'Адрес:' (g. Kamyslov ul. Maksima Gorkogo), 'Общежитие:' (нет), 'Группа:' (2П), and 'Дата зачисления:' (01.09.2020). A blue 'Сохранить' (Save) button is at the bottom right.

Рисунок 27 - Окно редактирования записи

Удалить запись можно нажав на соответствующую кнопку (см. Рисунок 28). После нажатия на кнопку система оповестит пользователя об удалении студента.

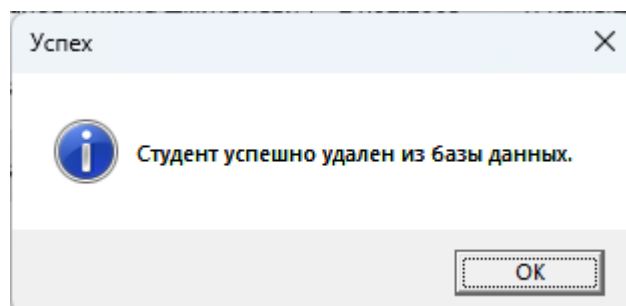


Рисунок 28 - Сообщение об удалении студента

Во всех окнах системы функции добавления, удаления и изменения реализованы в одном окне, что и просмотр записей.

## **Реализация функций поиска и фильтрации данных**

В приложении предусмотрены функции поиска, фильтрации и сортировки данных в окнах с таблицами базы данных для удобства и быстрого нахождения информации [34].

Алгоритм фильтрации данных в всех окнах приложения одинаков: необходимо выбрать параметр фильтрации из списка, и таблица будет обновлена в режиме реального времени в соответствии с выбранным критерием. Чтобы вернуться к полному списку, достаточно нажать кнопку "Восстановить".

Алгоритм поиска данных также аналогичен во всех окнах: необходимо ввести интересующее значение в поле поиска, и таблица будет обновлена в режиме реального времени, отображая только те строки, которые содержат введенный текст. Чтобы вернуться к полному списку, достаточно удалить текст из поля поиска.

Например, на странице студентов можно фильтровать список студентов по ФИО, для этого в текстовое поле вводятся данные студента (см. Рисунок 29)

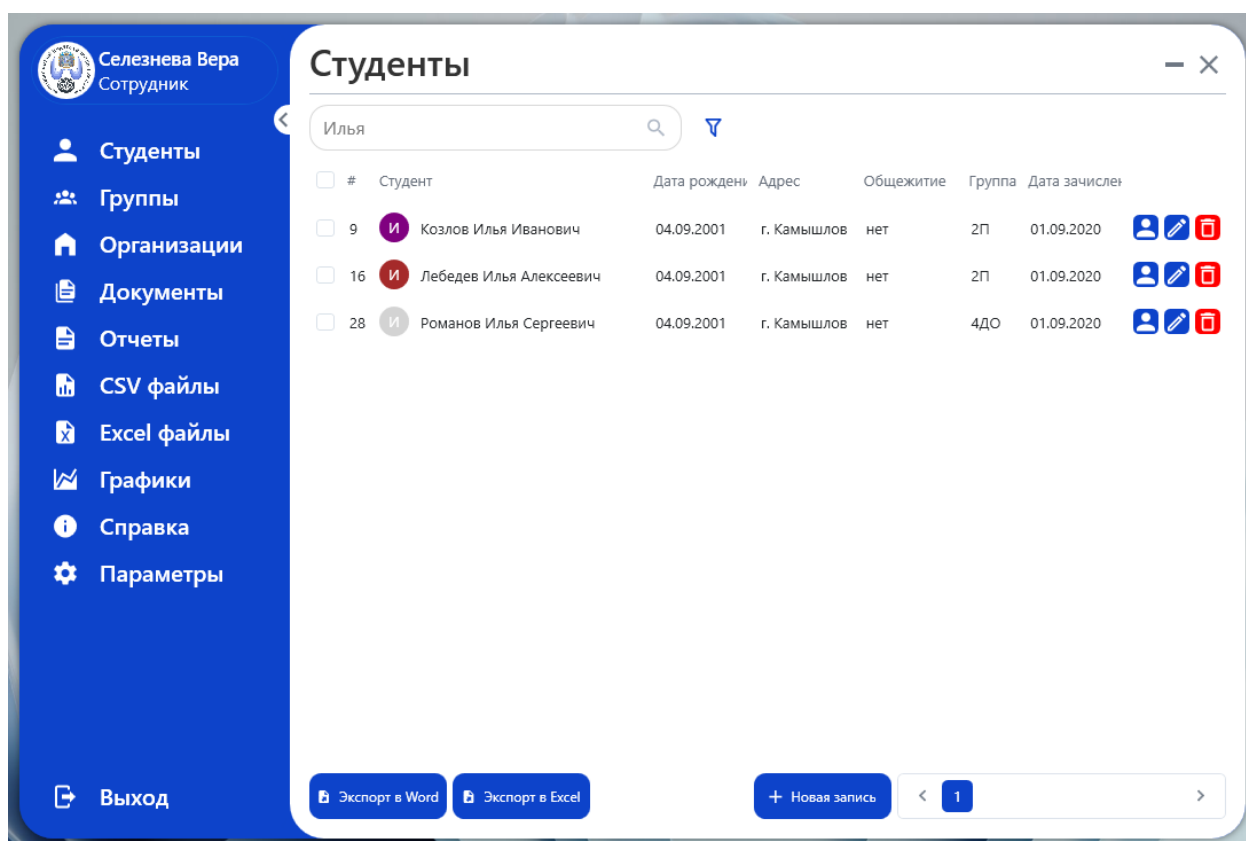


Рисунок 29 - Демонстрация фильтрации данных через текстовое поле

Для того что бы отфильтровать данные по группе необходимо выбрать группу из списка и нажать на кнопку «Поиск» (см. Рисунок 30).

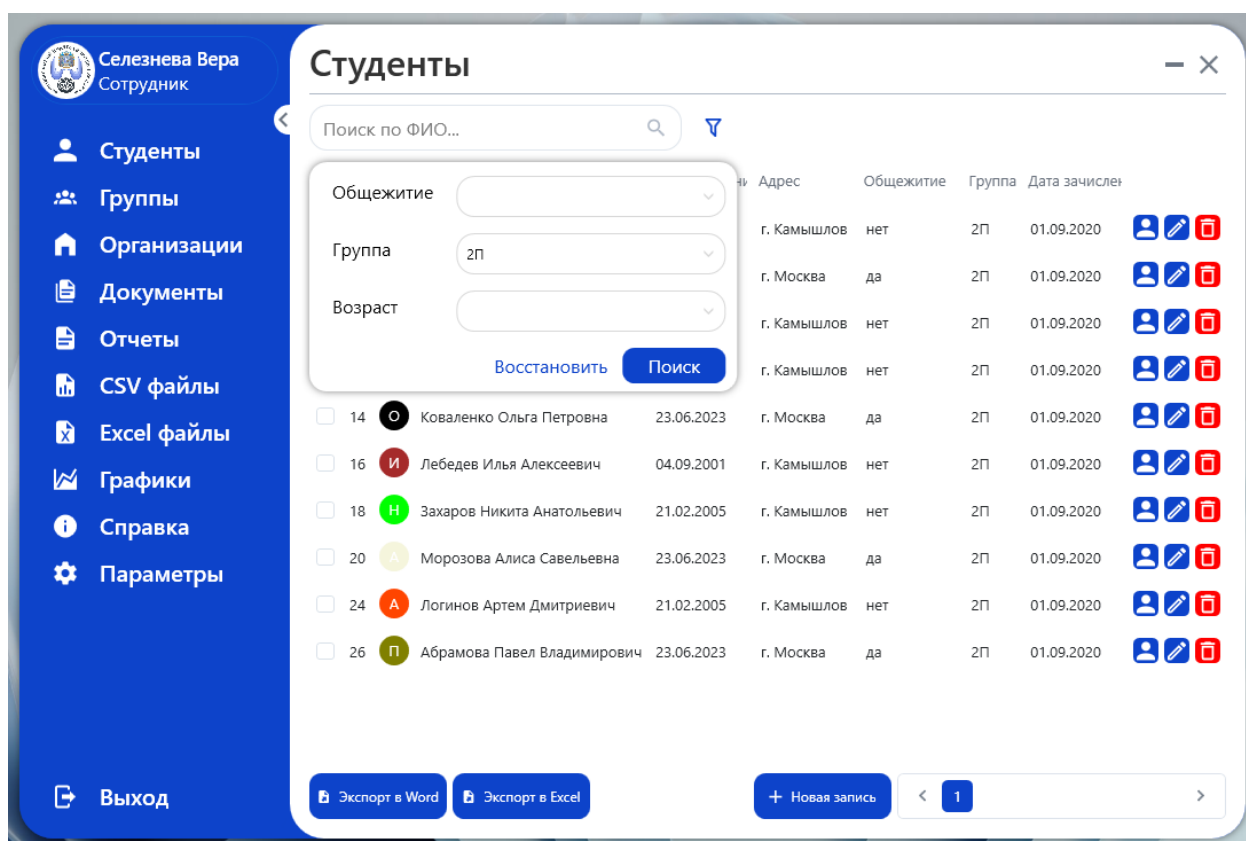


Рисунок 30 - Демонстрация фильтрации данных через выпадающий список

Для того что бы сбросить фильтрацию необходимо нажать на кнопку «Восстановить».

### Реализация функций просмотра и генерации отчетов

В приложении была успешно реализована функция просмотра и генерации отчетов для отчетности. Функция генерации отчетов позволяет создавать отчеты в различных форматах, таких как Excel и Word, на основе данных, извлеченных из базы данных.

Чтобы получить нужный отчет, перейдите на страницу "Отчеты" (см. Рисунок 31), выберите медицинский документ, по которому необходимо сформировать отчет, и нажмите соответствующую кнопку. После этого откроется программа Word с готовым отчетом (см. Рисунок 32).

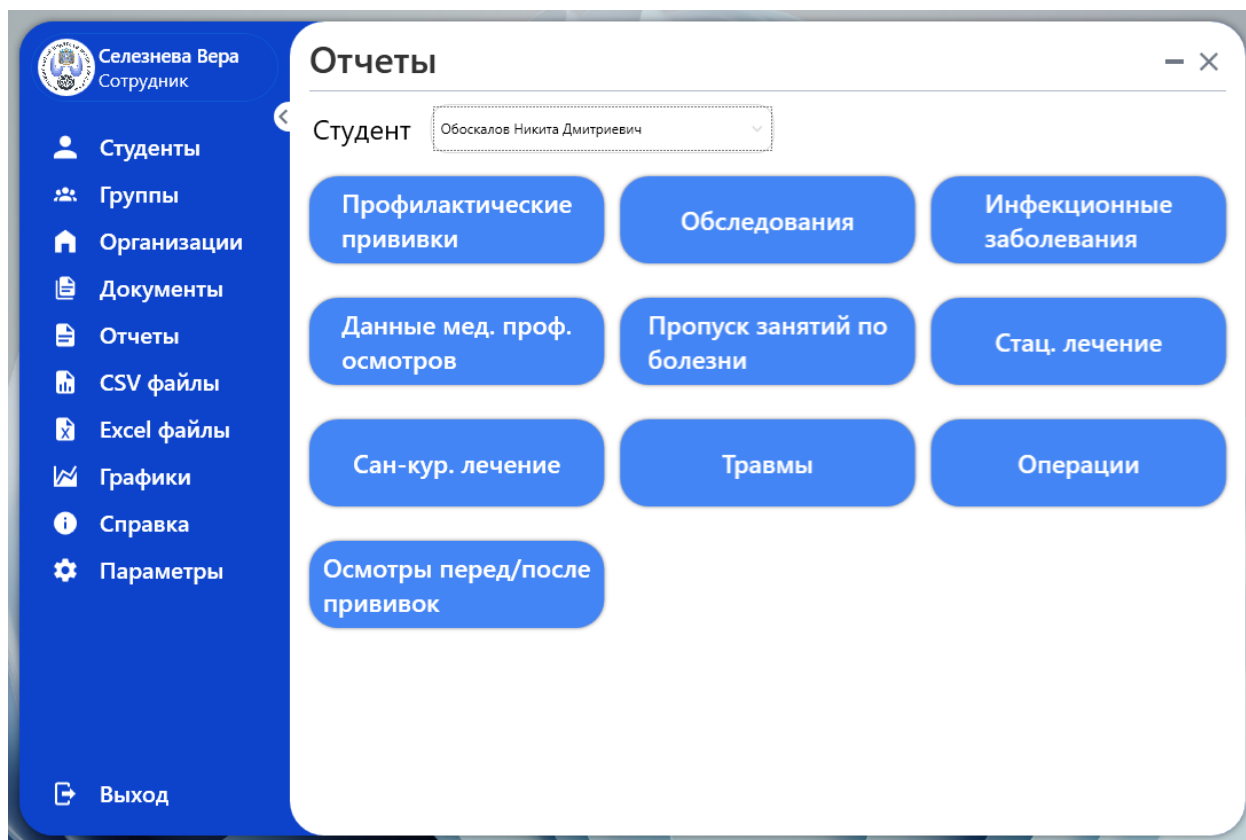


Рисунок 31 -Страница с отчетами

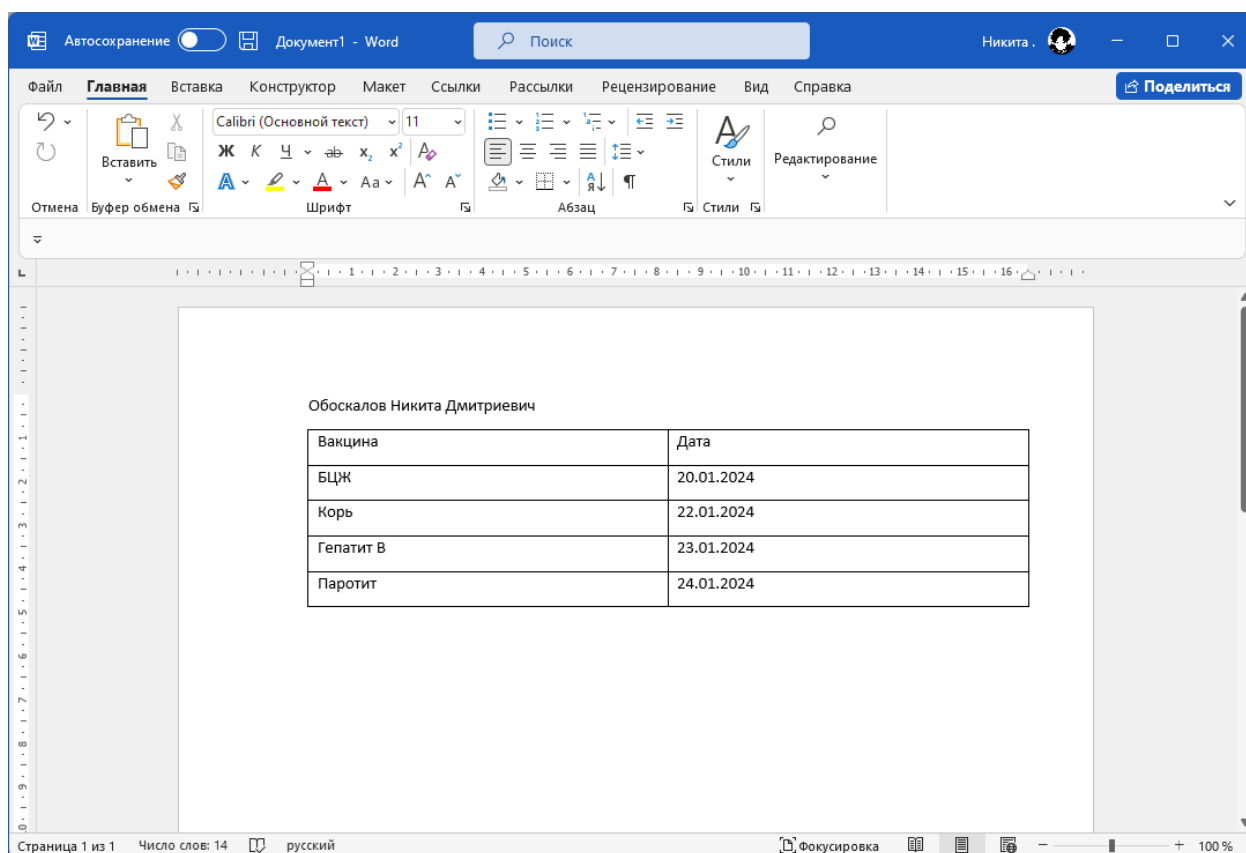


Рисунок 32 -Демонстрация готового отчета



## Реализация дополнительных функций приложений

В автоматизированной информационной системе предусмотрены дополнительные функции, облегчающие работу с данными. Одной из таких функций является постраничный просмотр списка (см. Рисунок 33). Эта функция позволяет выводить информацию для пользователя в определенном количестве, например, по тридцать записей на странице. Для переключения между страницами предусмотрены специальные кнопки в интерфейсе. Рядом с кнопками отображается максимальное количество страниц и номер текущей страницы, на которой находится пользователь.

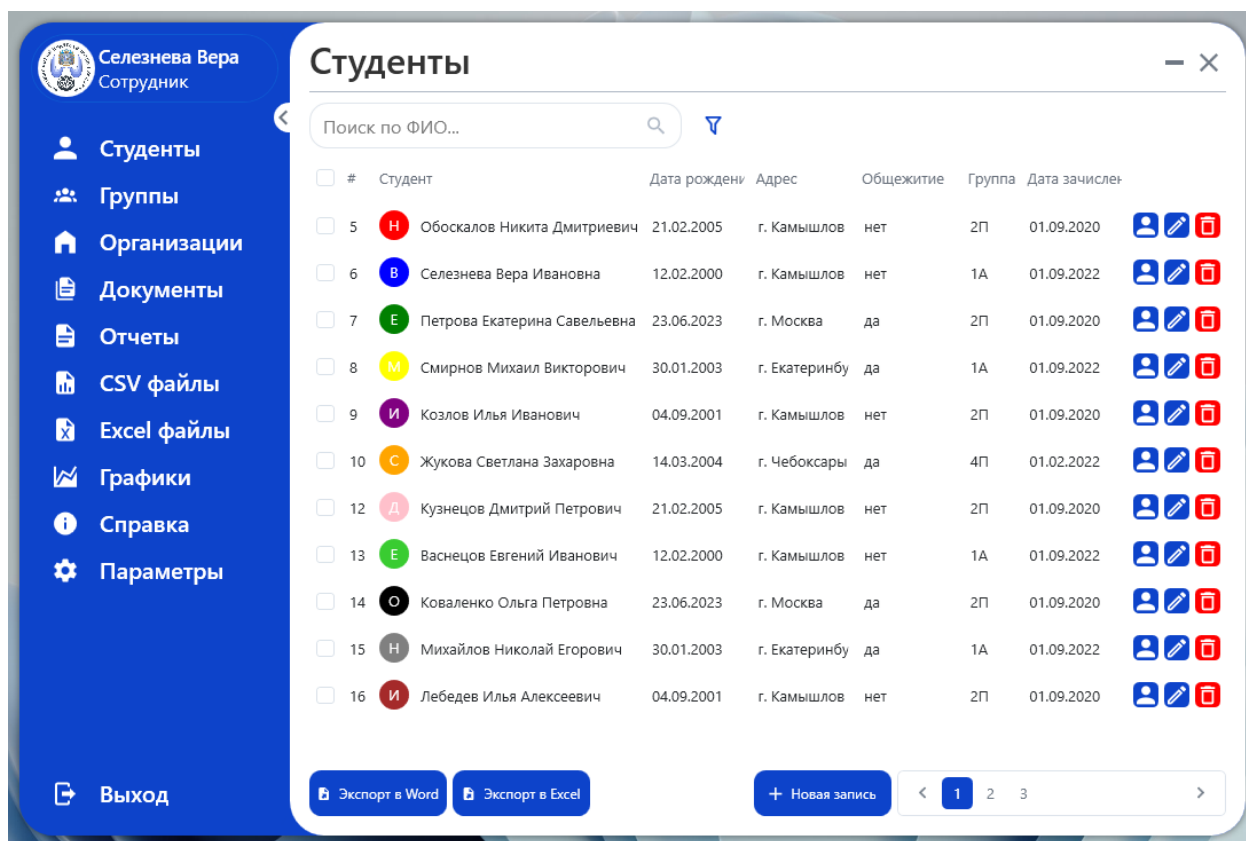


Рисунок 33 - Использование постраничного просмотра

Следующая дополнительная функция АИС: представление данных в виде графиков. Такой способ даёт возможность рассмотреть данные

графически. Удалось реализовать график, содержащий статистику об количестве человек в группе (см. Рисунок 34).

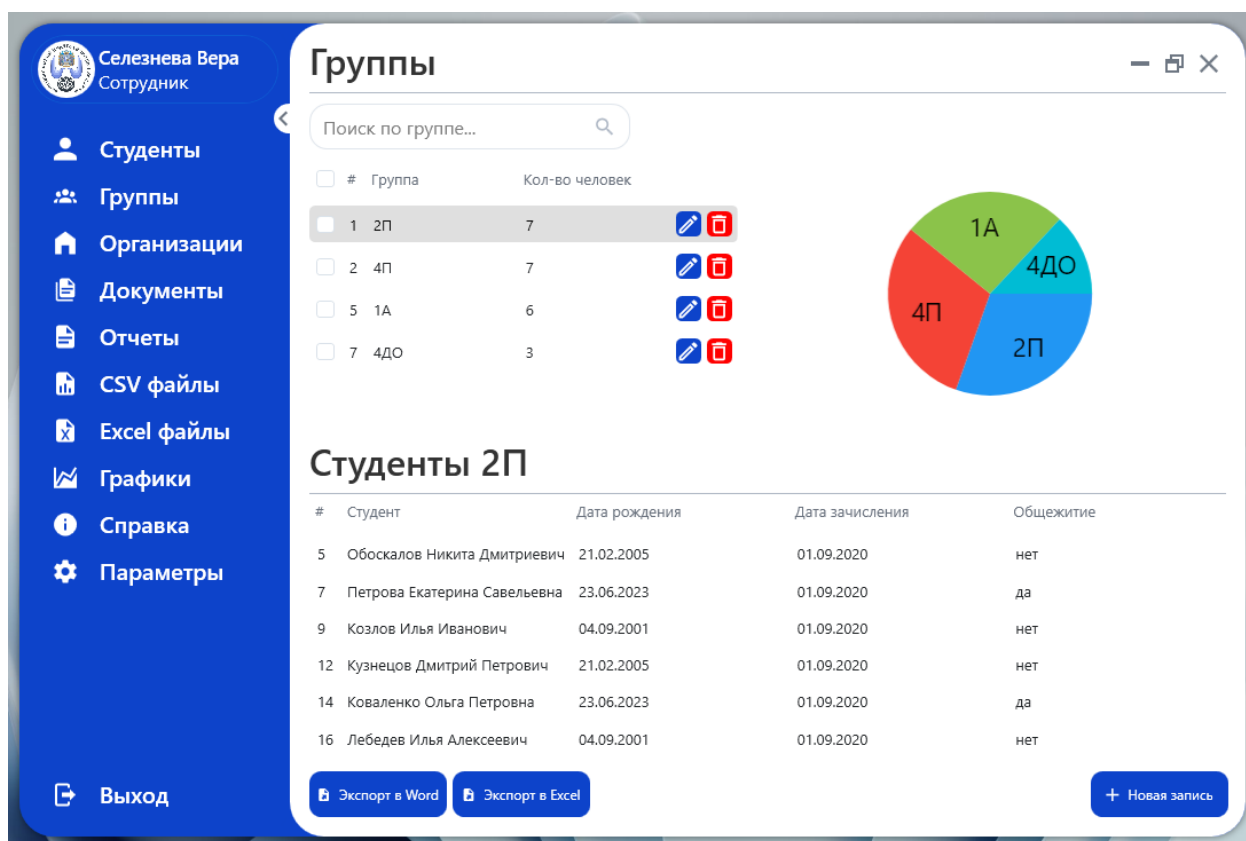


Рисунок 34 - Круговой график количества студентов в группах

График в виде круга представляет собой диаграмму, которая демонстрирует процентное соотношение различных категорий данных. В нашем случае, каждая категория представляет собой одну группу людей, а величина сектора круга соответствует количеству человек в этой группе.

В следующей функциональной возможности АИС предусмотрена запись файлов Excel. При создании этой функции было учтено многообразие форматов Excel-документов. Эта функция позволяет пользователям сохранять данные в файлы Excel, не покидая приложение. Запись означает сохранение данных в документ Excel.

## Тестирование приложений и отладка программных модулей

Отладка является важным этапом разработки программного обеспечения, который помогает выявлять и исправлять ошибки в коде. Инструменты отладки для приложений на языке C# с использованием технологии WPF и базы данных SQL Server:

1. Отладчик - это инструмент для поиска и исправления ошибок в коде, который позволяет программистам следить за выполнением программы шаг за шагом, устанавливать точки останова и просматривать значения переменных.

2. Логирование - это важный инструмент для отслеживания выполнения программы и выявления ошибок, который позволяет программистам выводить сообщения в логи для отслеживания последовательности выполнения программы и записи значений переменных или другой важной информации.

3. Профилирование - это процесс измерения производительности приложения с целью выявления узких мест и оптимизации кода, для которого используются инструменты типа Profiler, позволяющие анализировать производительность приложения и выявлять участки кода, где программа тратит больше всего времени или ресурсов.

Для отладки была выбрана форма авторизации пользователей, поскольку она включает в себя процесс аутентификации и авторизации, который часто требует работы с конфиденциальными данными пользователей. Отладка этой формы позволяет проверить правильность реализации класса шифрования и его взаимодействие с другими компонентами приложения. Это важно, потому что неправильная работа шифрования может привести к утечке конфиденциальной информации или уязвимостям безопасности.

Отладка окна авторизации с применением класса шифрования (см. Рисунок 35).

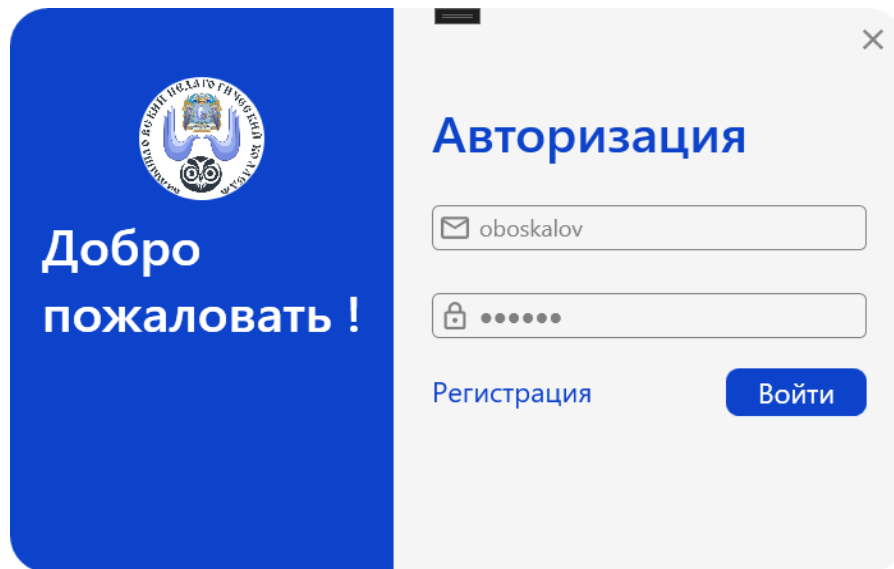


Рисунок 35 - Окно авторизации

Поставил точку останова в коде (см. Рисунок 36).

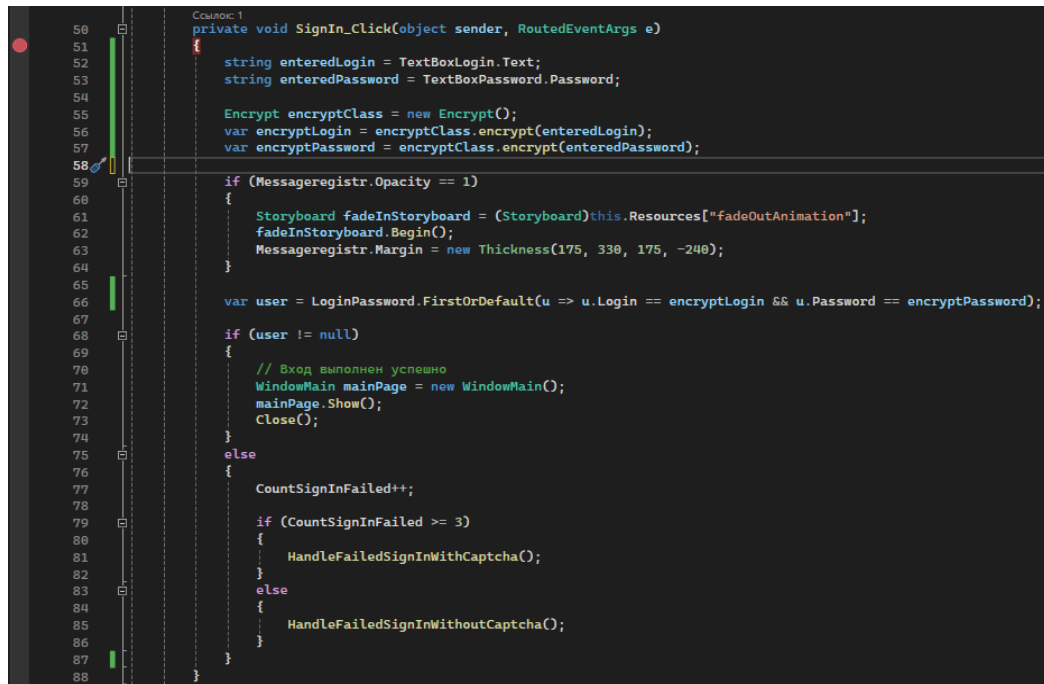


Рисунок 36 - Точка останова в коде шифрования

Запустил программу, ввел данные и нажал кнопку «Войти», после чего перекинуло на код, в котором стояла точка останова (см. Рисунок 37).

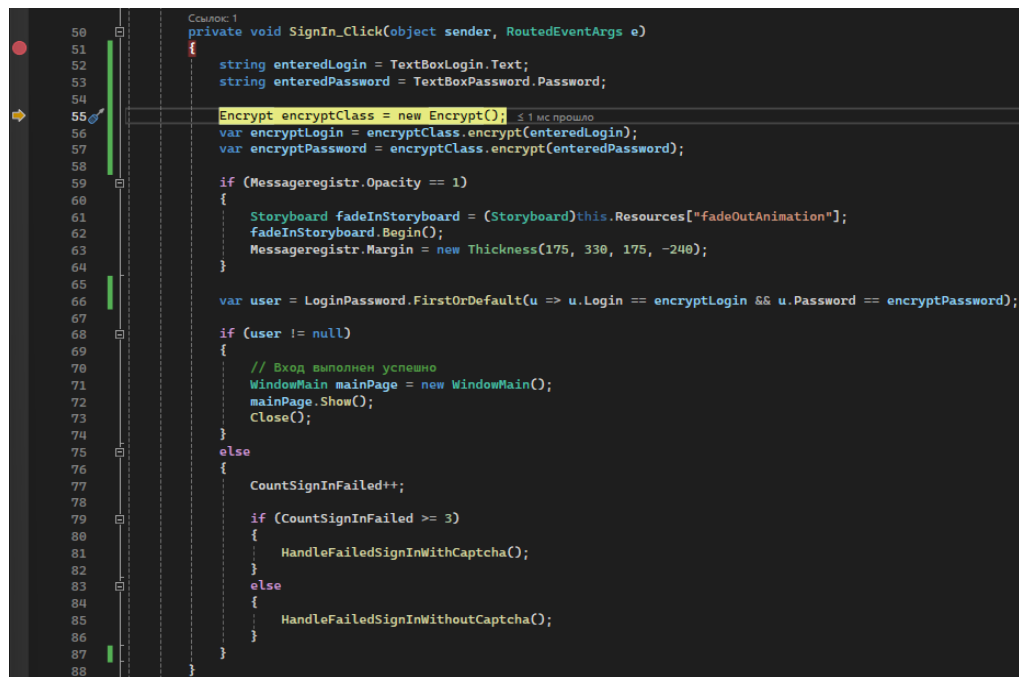


Рисунок 37 - Пошаговая отладка класса шифрования

После использования класса шифрования, вывел значения в окно контрольных значений (см. Рисунок 38).

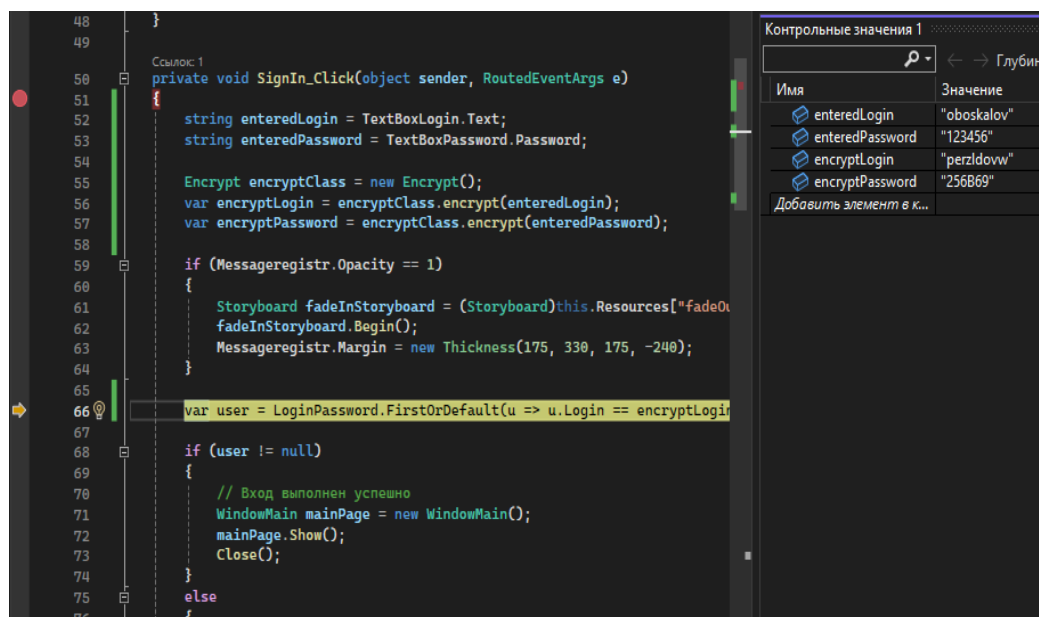


Рисунок 38 – Контрольные значения отладки класса шифрования

После использования класса шифрования и установки точки останова в соответствующем месте кода, я убедился, что программа корректно принимает значения из полей логина и пароля. Затем, с помощью класса шифрования, введенные пользователем данные были зашифрованы. Для проверки

корректности работы шифрования и сравнения значений с базой данных, были выведены значения в окно контрольных значений.

Таким образом, успешное выполнение этапов отладки окна авторизации с использованием класса шифрования подтвердило правильность его реализации и работу механизмов аутентификации в приложении.

Протестировал страницу добавления студента по классам эквивалентности входных данных в соответствии с требованиями заказчика (см. Рисунок 39), используя различные ограничения при вводе данных (см. Таблица 3).

Рисунок 39 - Окно добавления студента

Таблица 3 - Классы эквивалентности входных данных страницы добавление студента

Входные условия	Классы эквивалентности	
	Правильные	Неправильные
Значение поля «Фамилия» на странице добавление студента	Текстовое значение от 1 до 250 символов (1)	-не введено (2) -превышен лимит символов (3)

Значение поля «Имя» на странице добавление студента	Текстовое значение от 1 до 250 символов (4)	-не введено (5) -превышен лимит символов (6)
Значение поля «Отчество» на странице добавление студента	Текстовое значение от 1 до 250 символов (7)	-не введено (8) -превышен лимит символов (9)
Значение поля «Дата рождения» на странице добавление студента	Значение с 1.1.1900 по сегодняшнее (10)	-не выбрано (11) -будущая дата (12)
Значение поля «Проживание в общежитии» на странице добавление студента	Значения “да” или “нет” (13)	-не введено (14) -введены другие значения (15)
Значение поля «Группа» на странице добавление студента	Текстовое значение от 1 до 10 символов (19)	-не введено (20) -превышен лимит символов (21)
Значение поля «Дата зачисления» на странице добавление студента	Значение с 1.1.1900 по сегодняшнее (10)	-не выбрано (11) -будущая дата (12)

#### Тест для правильных классов эквивалентности

Заполнение полей страницы добавление студента правильными значениями (охватывает классы 1, 3, 6, 9, 12, 15, 19, 22, 25, 28, 31, 34):

Фамилия: Иванов

Имя: Иван

Отчество: Иванович

Дата рождения: 01.03.2024

Адрес: г. Камышлов

Проживание в общежитие: нет

Группа: 2П

Дата зачисления: 01.03.2024

Результаты: при нажатии кнопки «Сохранить» введенный студент добавится в базу данных и будет выведено в окно для просмотра, при открытии диалогового окна с возможностью выбора студента будет возможность выбора только что добавленного ученика.

#### Тесты для неправильных классов эквивалентности

Заполнение полей страницы добавление студента неправильными значениями (охватывает классы 2, 4, 7, 10, 11, 13, 14, 16, 17, 18, 20, 21):

Фамилия: Иванов

Имя: Иван

Отчество: Иванович

Дата рождения: 01.03.2024

Адрес: г. Камышлов

Проживание в общежитие: нет

Группа: 2П

Дата зачисления: 01.03.2024

Результаты: при нажатии на кнопку «Сохранить» выведется сообщение об ошибке «Заполните все поля». При закрытии программы или обновлении страницы все введенные данные теряются.

При тестировании страницы добавления студента по классам эквивалентности входных данных в соответствии с требованиями заказчика были применены различные ограничения при вводе данных. Это включало в себя проверку на минимальную и максимальную длину полей, а также наличие специфических символов или форматов данных, если они были указаны в требованиях. В результате тестирования было установлено соответствие функционала страницы требованиям заказчика, а также выявлены и исправлены потенциальные уязвимости или ошибки в обработке данных.

### **Создание инсталлятора программного обеспечения**

Программный инсталлятор - это специальная программа, предназначенная для установки программного обеспечения на компьютер пользователя.

После завершения разработки программного обеспечения был создан инсталлятор (см. Рисунок 40). С помощью инсталлятора можно выбрать место установки программы. После установки программного обеспечения на рабочем столе пользователя появится ярлык для быстрого запуска.



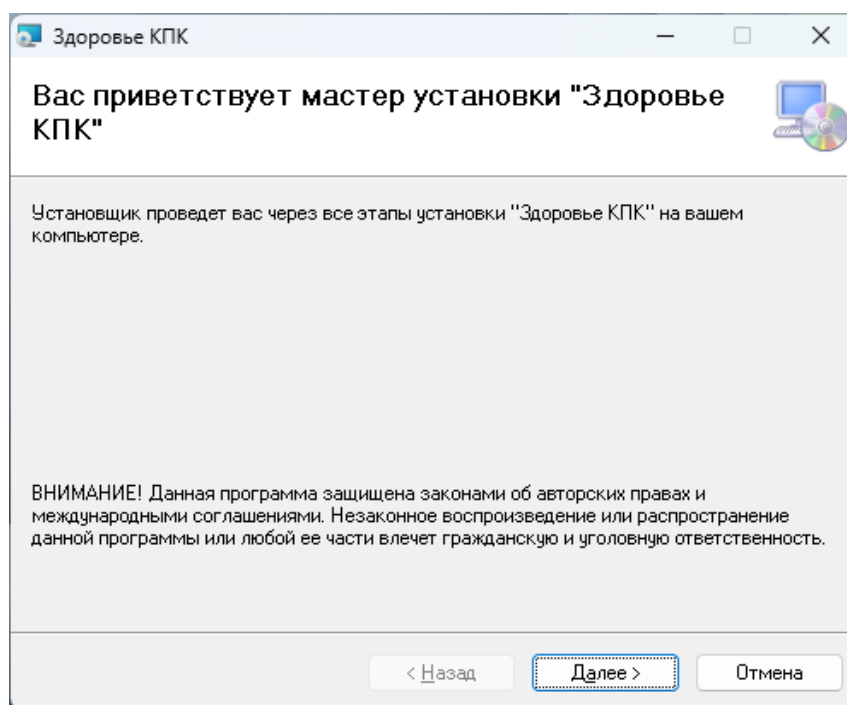


Рисунок 40 - Инсталлятор программного обеспечения

## Разработка сопровождающей технической документации

Сопровождающая техническая документация — это своего рода документация на программное обеспечение, представляющая из себя печатные и электронные руководства пользователя, справочный текст, описывающие как пользоваться программным продуктом [8].

После создания технической документации для автоматизированной информационной системы, было принято решение интегрировать руководство пользователя непосредственно в приложение [13]. Это руководство содержит подробную информацию о функциях программы и правилах ее использования.

Чтобы воспользоваться документацией, пользователю необходимо перейти на страницу «Справка» и выбрать необходимый пункт (см. Рисунок 41). После этого откроется документ в формате PDF, в котором пользователь может найти необходимую информацию о работе с программой. Все разделы руководства подробно описывают соответствующие функции и дают практические рекомендации по их использованию.

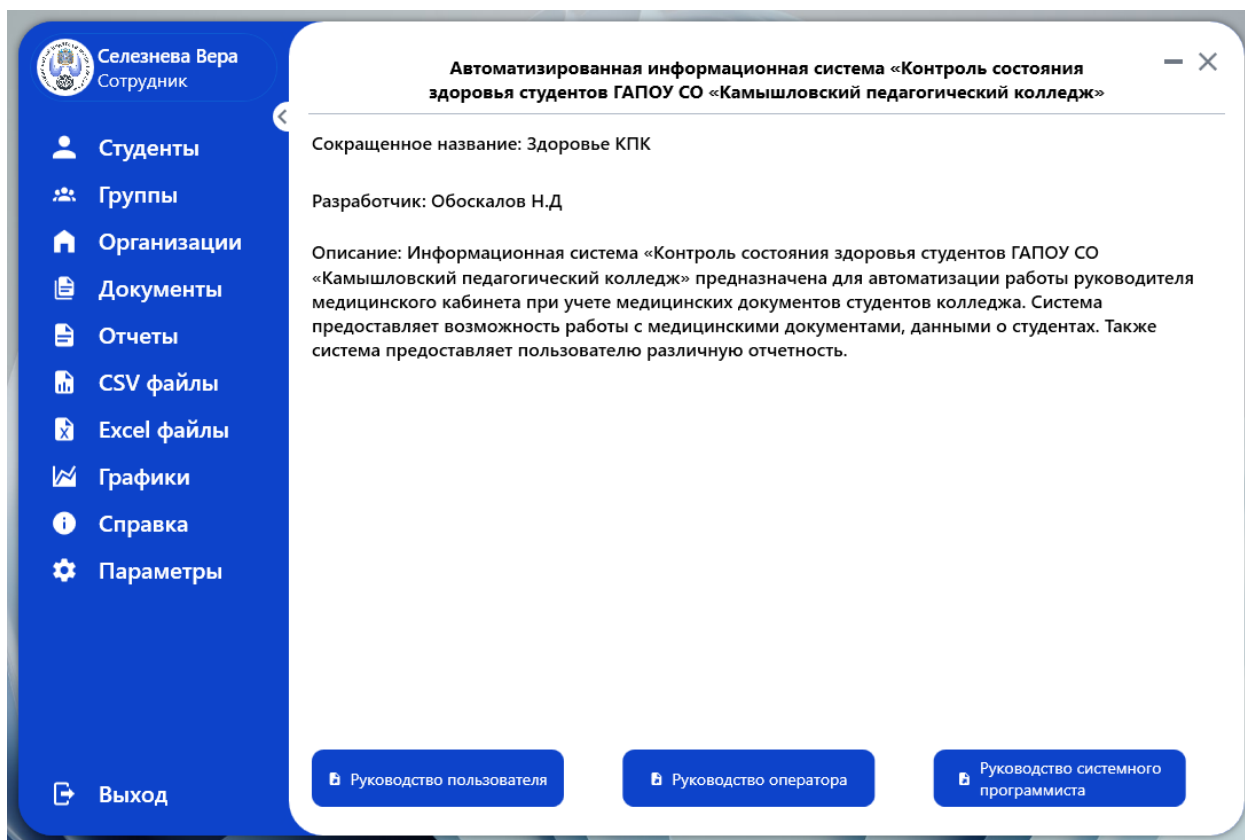


Рисунок 41 - Страница «Справка»

## Оценка экономической/социальной эффективности проекта

Единовременные (капитальные) затраты вычисляются по следующей формуле:

$$K = K_{\text{пр}} + K_{\text{тс}} + K_{\text{лс}} + K_{\text{по}} + K_{\text{ио}} + K_{\text{об}} + K_{\text{оэ}}$$

где  $K_{\text{пр}}$  – затраты на проектирование;

$K_{\text{тс}}$  – затраты на технические средства управления;

$K_{\text{лс}}$  – затраты на создание линий связи локальных сетей;

$K_{\text{по}}$  – затраты на программные средства;

$K_{\text{ио}}$  – затраты на формирование информационной базы;

$K_{\text{об}}$  – затраты на обучение персонала;

$K_{\text{оэ}}$  – затраты на опытную эксплуатацию.

Затраты на проектирование рассчитываются следующим образом:

$$K_{\text{пр}} = K_{\text{зп}} + K_{\text{инс}} + K_{\text{свт}} + K_{\text{проч}}$$

где  $K_{\text{зп}}$  – затраты на заработную плату проектировщиков;

$K_{\text{инс}}$  – затраты на инструментальные программные средства для проектирования;

$K_{\text{свт}}$  – затраты на средства вычислительной техники для проектирования;

$K_{\text{проч}}$  – прочие затраты на проектирование.

В рамках разработки дипломного проекта единовременные затраты сведены к нулю, так как разработка автоматизированной информационной системы не подразумевает какие-либо дополнительные покупки.

Так как для разработки автоматизированной информационной системы использовались инструментальные программные средства, то были выполнены соответствующие затраты для проектирования (см. Таблица 4).

Таблица 4 - Затраты на инструментальные программные средства для проектирования.

Программное средство	Стоимость
Microsoft Visual Studio Community 2019 Версия 16.8.5 © Корпорация Майкрософт (Microsoft Corporation), 2020	0,00 Р
SQL Server Management Studio v18.8 © Корпорация Майкрософт (Microsoft Corporation), 2020	0,00 Р
Microsoft® Office 2016 MSO (16.0.4266.1001), 64-разрядная версия.	0,00 Р
SQL Server Management Studio (SSMS) v18.8	0,00 Р
Итого:	0,00 Р

Прочие затраты на проектирование составляют 0% от совокупных затрат на проектирование.

Полная стоимость затрат на проектирование составляет:

$$K_{\text{пр}} = \text{з/пл} + \text{затр на инстр ПС} + \text{прочие затраты}$$

Затраты на технические средства управления составляют 0 руб, так как нет необходимости в их приобретении. Амортизационные отчисления за время проектирования очень малы.

Затраты на формирование информационной базы включают в себя:

Заработную плату администратора, программное обеспечение для работы с базой данных и затраты на ее хранение. Стоимость хранения базы данных равняется нулю так как база данных будет использоваться лишь в локальной сети и не нуждается в хостинге, поэтому в данных затратах не берется в расчет. Программное обеспечение, которое использовалось для работы с базой данных, является бесплатным (см. Таблица 5).

Затраты на обучение персонала составляют  $0 = \text{з/п} / \text{кол-во раб дней} = 0$  руб. В рамках разработки дипломного проекта затраты на обучение персонала не предоставляются.

Затраты на опытную эксплуатацию сведены к нулю, так как в этом нет необходимости.

Таблица 5 - Единовременные и прочие затраты

Вид затрат	Капитальные затраты	
	Состав затрат	Планируемая сумма
Единовременные затраты	Microsoft Visual Studio Community 2019 Версия 16.8.5 © Корпорация Майкрософт (Microsoft Corporation), 2020	0,00 Р
	SQL Server Management Studio v18.8 © Корпорация Майкрософт (Microsoft Corporation), 2020	0,00 Р
	Microsoft® Office 2016 MSO (16.0.4266.1001), 64-разрядная версия.	0,00 Р
	SQL Server Management Studio (SSMS) v18.8	0,00 Р
Прочие затраты	Затраты на обучение персонала	0,00 Р
	Затраты на опытную эксплуатацию	0,00 Р

Оценка внедрения: разработка является наиболее эффективной с экономической точки зрения так как обслуживание и внедрение данной разработки не требует дополнительных затрат кроме программных средств и заработной платы администратора, работающего с автоматизированной информационной системой.

Ключевыми факторами социальной и экономической эффективности являются:

1. Минимизация упущенного дохода или формирование новых источников дохода;
2. Снижение текущих производственных (эксплуатационных) затрат;
3. Снижение административно-управленческих затрат;
4. Минимизация налоговых и других обязательных выплат;
5. Снижение потребности в капитальных затратах;
6. Увеличение оборачиваемости текущих активов.

Возможные статьи дохода от внедрения разработанного программного продукта в ГАПОУ СО «Камышловский педагогический колледж»:

— увеличение производительности работы медицинского кабинета колледжа: Разработанный программный продукт может улучшить эффективность ведения медицинской документации, позволив быстрее и точнее регистрировать данные студентов, отслеживать медицинские аспекты и проводить анализ состояния здоровья студентов.

— снижение затрат на медицинские услуги и лекарственные средства: благодаря более эффективному мониторингу состояния здоровья студентов и своевременной диагностике возможных проблем, колледж может снизить затраты на медицинские услуги и лекарственные средства за счет предотвращения возможных заболеваний и своевременного лечения.

— оптимизация учебного процесса: Улучшение доступа к информации о состоянии здоровья студентов может способствовать более эффективной организации учебного процесса. Это может привести к повышению успеваемости студентов, сокращению времени на административные процедуры и увеличению общей эффективности образовательной деятельности.

Эффектом внедрения информационной системы (ИС) в организацию является улучшение эффективности ее деятельности, оптимизация процессов

и повышение конкурентоспособности. Внедрение ИС может привести к следующим результатам:

- автоматизация и оптимизация бизнес-процессов: ИС позволяет автоматизировать рутинные операции, сокращает время выполнения задач, улучшает координацию работы между подразделениями и сокращает вероятность ошибок;

- улучшение доступа к информации: ИС обеспечивает быстрый и удобный доступ к необходимой информации для принятия решений на всех уровнях управления. Это позволяет сократить время на поиск и обработку данных;

- экономия времени и ресурсов: ИС помогает оптимизировать расходы на персонал, сокращает затраты на бумажную документацию, минимизирует потери информации и уменьшает количество ошибок;

- улучшение управленческих решений: ИС предоставляет руководителям точную и своевременную информацию для анализа ситуации и принятия обоснованных управленческих решений.

Существуют два типа методов оценки эффективности проекта: финансовый и качественный. В данной работе оценка будет производиться по методу информационной экономики (Information Economics), которая относится к качественным методам.

Следуя данному методу, в первую очередь необходимо определить цели, которые нужно достичь с помощью внедрения проекта:

- повышение производительности: в контексте медицинского кабинета колледжа это может означать увеличение скорости обработки медицинских данных, сокращение времени на административные процедуры, повышение эффективности работы персонала и оптимизацию использования ресурсов;

- сокращение времени выполнения административных процедур: целью может быть сокращение времени, затрачиваемого на оформление

документов, ведение отчетности и другие административные процедуры, связанные с медицинским кабинетом колледжа. Это поможет освободить время персонала для более важных задач и улучшить общую эффективность работы;

— улучшение доступности и точности медицинских данных: целью может быть обеспечение легкого доступа к медицинской информации о студентах для врачей и других уполномоченных лиц, а также повышение точности и достоверности этой информации. Это поможет улучшить качество медицинского обслуживания студентов и снизить риск ошибок;

— увеличение безопасности и конфиденциальности данных: целью может быть обеспечение надежной защиты медицинских данных студентов от несанкционированного доступа и утечек информации. Это поможет соблюдать требования по защите конфиденциальности и приватности данных, а также повысит доверие студентов к колледжу;

— сокращение рисков и повышение ответственности: целью может быть сокращение рисков возникновения медицинских ошибок, неправильного ведения медицинской документации и других негативных событий. Повышение ответственности перед студентами и их родителями за состояние здоровья и безопасность также может быть важной целью внедрения информационной системы.

Реализованный программный продукт удовлетворяет всем поставленным целям.

Перечень возможных рисков проекта:

— Технические риски: проблемы совместимости с существующими информационными системами колледжа. Трудности при интеграции новой информационной системы с другими учебными или административными системам;

— Риски безопасности: угрозы кибербезопасности, такие как хакерские атаки или вирусы, которые могут угрожать конфиденциальности

медицинских данных студентов. Недостаточная защита данных от несанкционированного доступа или утечки информации;

— Организационные риски: сопротивление персонала изменениям или нехватка поддержки со стороны управления колледжа. Недостаточная подготовка персонала к использованию новой информационной системы;

— Финансовые риски: превышение бюджета проекта из-за непредвиденных затрат или недостаточной оценки стоимости реализации. Неэффективное использование ресурсов или перерасход средств на проекте;

— Технологические риски: недостаточная надежность или производительность используемого программного обеспечения. Отсутствие поддержки или обновлений для выбранного программного обеспечения в будущем;

— Людские риски: уход ключевых специалистов, работающих над проектом, что может привести к задержкам в разработке или потере ценной экспертизы. Недостаточная коммуникация и сотрудничество между участниками проекта, что может замедлить процесс реализации.

Провел качественную оценку рисков. Риски оцениваются по трехуровневой шкале влияния рисков на реализацию проекта:

— «низкий уровень» – малый риск, незначительное влияние на проект;

— «средний уровень» – риск, имеющий равную вероятность повлиять и не повлиять на результат исполнения проекта;

— «высокий уровень» – высокий риск, заключающийся в неисполнении проекта или в значительном изменении результата его реализации.

Приведены результаты качественной оценки рисков (см. Таблица 6). Также указаны мероприятия по снижению вероятности рисков, имеющих наивысшую оценку.

Таблица 6 - Риски проекта



<b>Риск</b>	<b>Потери</b>	<b>Уровень влияния риска на проект</b>	<b>Возможность предотвращения или снижения риска</b>
Недостаточная квалификация	Ненамеренное повреждение файлов сотрудниками организации	«средний уровень»	Самообразование
Недостаточная защита информационных ресурсов	Утечка конфиденциальной информации	«высокий уровень»	Предотвращение утечки, хищения, утраты, искажения, подделки информации
	Взлом веб-приложения злоумышленниками;		
Сбои в работе системы	Повышенная нагрузка, ведущая к снижению результатов.	«высокий уровень»	Перезапуск автоматизированной информационной системы

Объектом проведенного исследования является рабочее место сотрудника медицинского кабинета ГАПОУ СО «Камышловский педагогический колледж».

Преимущества использования специалистом разработанного продукта заключаются в:

— снижение затрат: программный продукт, разработанный студентами, часто может быть доступен по более низкой цене или даже бесплатно, что снижает затраты на приобретение программного обеспечения для организации;

— инновационные решения: студенты, работающие над дипломными проектами, часто вносят свежие идеи и инновационные решения, которые могут быть полезны для организации. Использование продукта, созданного студентами, может способствовать внедрению новых подходов и методов в работу организации;

— поддержка образовательных инициатив: поддержка программ дипломной практики и научно-исследовательских проектов студентов способствует развитию образовательной среды и сотрудничеству между образовательными учреждениями и бизнесом;

— адаптивность и гибкость: программный продукт, созданный студентами, может быть более адаптивным и гибким к потребностям организации, так как он может разрабатываться с учетом специфики ее бизнес-процессов и требований;

— развитие потенциала персонала: участие в разработке программного продукта во время дипломной практики может быть ценным опытом для студентов и способствовать их профессиональному развитию. В то же время, использование такого продукта позволяет организации развивать потенциал молодых специалистов и привлекать их в будущем на работу.

Риски при использовании продукта заключаются в следующем:

— недостаточное качество продукта: программный продукт может оказаться низкого качества из-за ошибок в разработке, несоответствия требованиям пользователя или недостаточного тестирования перед внедрением;

— проблемы совместимости: продукт может оказаться несовместимым с другими программными системами или оборудованием, что может затруднить его интеграцию в существующую инфраструктуру организации;

— отсутствие поддержки и обновлений: если разработчики продукта не обеспечивают его поддержку и регулярные обновления, это может привести к проблемам с безопасностью, исправлением ошибок и адаптацией к изменяющимся потребностям организации;

— недостаточная защита данных: программный продукт может оказаться уязвимым к кибератакам или не обеспечивать достаточной защиты данных, что может привести к утечкам информации или потере конфиденциальности;

— неэффективное использование: если пользователи не будут правильно обучены или не увидят ценности в использовании программного продукта, это может привести к его неполному или неэффективному

использованию, что снизит общую эффективность и рентабельность внедрения;

— недостаточная гибкость и адаптивность: программный продукт может оказаться недостаточно гибким или неспособным адаптироваться к изменяющимся потребностям организации или бизнес-процессам, что снизит его ценность на длительную перспективу.

С учетом вышеописанных преимуществ, использование разработанного продукта является эффективным и обоснованным. Продукт обладает рядом преимуществ, которые способствуют повышению эффективности работы организации и достижению ее целей.

Выводы по разделу «Экономическая и социальная эффективность от внедрения.

В данном разделе произведен расчет всех капитальных (единовременных) и эксплуатационных затрат, а также совокупной стоимости владения информационной системой (ИС).

Далее произведена оценка внедрения ИС как стратегического проекта, в рамках которого учитываются его влияние на бизнес-процессы, цели и стратегические планы организации.

Также составлен перечень возможных рисков проекта, проведена качественная оценка и указаны мероприятия по снижению вероятности рисков и их влияния на успешное внедрение и эксплуатацию ИС.

Все полученные данные представлены в сводных таблицах и диаграммах для наглядного и комплексного анализа и принятия решений о дальнейших шагах в реализации проекта.

## **Заключение**

При выполнении дипломного проекта были определены цели, задачи и требования к разрабатываемой информационной системе. Был проведен анализ предметной области, выбраны и проанализированы методы решения. Было выполнено проектирование информационной системы и приложения конечных пользователей, включая выбор среды разработки приложения, проектирование с использованием UML диаграмм, методологии SADT, методологии IDEF1X, проектирование пользовательского интерфейса. Были изучены различные методологии проектирования баз данных и успешно спроектирована база данных. Были реализованы ограничения по спецификациям на ввод данных и созданы необходимые объекты.

В результате была разработана информационная система, которая включает в себя разработанный интерфейс, функции поиска по различным критериям, фильтрацию данных, просмотр и генерацию отчетов, а также дополнительные функции, такие как агрегатные функции, статистика, вычисление штрафов, графики и диаграммы по информации из базы данных. Основными функциями системы являются функции добавления, редактирования и удаления записей, которые реализованы на каждой форме вывода информации. Пользователи также имеют возможность просматривать документы в форматах Excel, Word и других, что обеспечивает удобство работы с различными типами документов прямо в приложении.

Реализована функция шифрования данных, которая используется при регистрации новых пользователей в системе. Благодаря этому, информация о пароле и логине пользователя защищена от несанкционированного доступа. Такой подход позволяет обеспечить безопасность пользовательских данных и предотвратить возможные утечки информации.

Выполнено тестирование приложений и отладка программных модулей для обнаружения ошибок и сбоев в программе. Разработана сопровождающая техническая документация для обеспечения подробного описания продукта,

включая характеристики, функционал, принципы работы и инструкции по использованию. Проведена оценка экономической и социальной эффективности проекта ключевые параметры, оказывающие влияние на успешность проекта.

В приложении организовала просмотр информации с соединением нескольких таблиц с помощью запросов по требованиям пользователя. Так же просмотр информации связанных таблиц. Реализовано экспорт информации в виде отчетов, которые можно распечатать на бумаге или сохранить в виде файлов Excel, Word. Для удобства пользоваться программой разработано руководство для пользователя. В котором прописаны функциональные возможности, описание о каждом разделе в приложении.

Таким образом программа предназначена для автоматизации и облегчения работы сотрудника медицинского кабинета. Простой интерфейс программы позволяет понять принцип ее работы и максимально использовать все ее функции. Данная автоматизированная информационная система должна работать с оперативными данными, накопление этих данных позволит проводить анализ деятельности за любой период времени. Это является одной из задач внедрения системы. Благодаря автоматизированной информационной системе для сотрудника медицинского кабинета работа по контролю здоровья студентов существенно сокращает время, затрачиваемое на подготовку информации для дальнейшего использования.

## Список литературы

1. Арлоу Д., Нейштадт И. UML 2 и Унифицированный процесс. Практический объектно-ориентированный анализ и проектирование, 2е издание. Пер. с англ. СПб: Символ Плюс, 2007. – 624 с.
2. Бахтизин В.В. Методология функционального проектирования IDEF0: Учеб. пособие по курсу «Технология разработки программного обеспечения» для студ. спец. 40 01 01 «Программное обеспечение информационных технологий дневной формы обучения» Мн.: БГУИР, 2003. – 24 с.
3. Бурцева Е.В. Информационные системы: учеб. пособие / И.П. Рак, А.В. Селезнев, А.В. Терехов, В.Н. Чернышов. Тамбов: Изд-во Тамб. гос. техн. ун-та, 2009. – 128 с. – 150 экз. – ISBN 978-5-8265-0874-9.
4. Васильков А. В. Безопасность и управление доступом в информационных системах / А.В. Васильков, И.А. Васильков. - М.: Форум, 2015. - 368 с.
5. Горбаченко В. И. Проектирование информационных систем с СА ERwin Modeling Suite 7.3: учебное пособие / В. И. Горбаченко, Г. Ф. Убиенных, Г. В. Бобрышева – Пенза: Изд-во ПГУ, 2012. – 154 с.
6. Горлушкина Н.Н. Системный анализ и моделирование информационных процессов и систем. СПб: Университет ИТМО, 2016. – 120 с.
7. ГОСТ 19.701-90 Единая система программной документации (ЕСПД). Схемы алгоритмов, программ, данных и систем (с Изменением N 1) от 1 сентября 2020 г. дата введения установлена 01.01.1992 ИЗДАНИЕ (январь 1991 г.) с Изменением N 1, утвержденным в сентябре 1990 г. (ИУС 11-81).
8. ГОСТ 19.106-78 Единая система программной документации (ЕСПД). Требования к программным документам, выполненным печатным способом (с Изменением N 1) от 18 декабря 1978 г. N 3350 дата введения

установлена 01.01.80 ИЗДАНИЕ (январь 2010 г.) с Изменением N 1, утвержденным в сентябре 1981 г. (ИУС 11-81).

9. Ручкин К.А., Ручкина В.Н. Моделирование бизнес-процессов с помощью современных информационных технологий // Менеджмент в России и за рубежом. -2007 - №1. - С. 132.

10. Базы данных. Разработка клиентских приложений на платформе Net: учебник / Н.Н. Гриченко, А.Ю. Громов, А.В. Благодаров. М.: КУРС, 2018. - 288 с.

11. Илюшечкин В. М. Основы использования и проектирования баз данных: учебник для среднего профессионального образования / В. М. Илюшечкин. — испр. и доп. — Москва: Издательство Юрайт, 2019. — 213 с. — (Профессиональное образование). — ISBN 978-5-534-01283-5. — Текст: электронный // ЭБС Юрайт [сайт]. — URL: <https://biblio-online.ru/bcode/437670> (дата обращения: 31.01.2020).

12. ГОСТ 19.201-78 Единая система программной документации (ЕСПД). Техническое задание. Требования к содержанию и оформлению (с Изменением N 1) от 18 декабря 1978 N 3351 дата введения установлена 01.01.80 ИЗДАНИЕ (январь 2010 г.) с Изменением N 1, утвержденным в июне 1981 г. (ИУС 9-81).

13. ГОСТ 2.301-68 Единая система конструкторской документации (ЕСКД). Форматы (с Изменениями N 1, 2, 3) от 28 мая 1968 ИЗДАНИЕ (август 2007 г.) с Изменениями N 1, 2, 3, утвержденными в декабре 1980 г., марте 1989 г., июне 2006 г. (ИУС 3-81, 7-89, 9-2006).

14. Грекул В. И. Проектирование информационных систем: учебник и практикум для среднего профессионального образования / В. И. Грекул, Н. Л. Коровкина, Г. А. Левочкина. — Москва: Издательство Юрайт, 2022. — 385 с. — (Профессиональное образование). — ISBN 978-5-534-12104-9. — Текст : электронный // Образовательная платформа Юрайт [сайт]. — URL: <https://urait.ru/bcode/496196> (дата обращения: 09.03.2022).

15. Зараменских Е. П. Информационные системы: управление жизненным циклом: учебник и практикум для среднего профессионального образования / Е. П. Зараменских. — Москва: Издательство Юрайт, 2022. — 431 с. — (Профессиональное образование). — ISBN 978-5-534-11624-3. — Текст: электронный // Образовательная платформа Юрайт [сайт]. — URL: <https://urait.ru/bcode/495987> (дата обращения: 09.03.2022).
16. Иванов Д. Ю., Новиков Ф. А. Основы моделирования на UML: Учеб. пособие. — СПб.: Изд-во Политехн. ун-та, 2010. — 249с.
17. Информационные системы - миф и действительность. М.: Знание, 2016. - 887 с.
18. Кара-Ушанов В.Ю. SQL — язык реляционных баз данных: учебное пособие / В.Ю. Кара-Ушанов. — Екатеринбург: Изд-во Урал. ун-та, 2016.— 156 с.
19. Карпова И. П. Базы данных. Курс лекций и материалы для практических заданий. — Учебное пособие. — М.: Питер, 2013. — 240 с.
20. Масленникова О.Е. Разработка реляционных баз данных с использованием CASE-средства ALL Fusion Data Modeler [Электронный ресурс] / О.Е. Масленникова, О.Б. Назарова. - М.: ФЛИНТА, 2016. - 74 с. - URL: <http://e.lanbook.com/> (ЭБС «ЛАНЬ») (дата обращения: 11.03.2022).
21. Маторин С.И. Теория систем и системный анализ: Учебное пособие / С.И. Маторин., О.А. Зимовец. — Белгород: Изд-во НИУ «БелГУ», 2012. — 288 с.
22. Медведев М. А. Программирование на СИ#: учеб. пособие / М. А. Медведев, А.Н. Медведев. — Екатеринбург: Изд-во Урал. ун-та, 2015. — 64 с.
23. Мезенцев К. Н. Автоматизированные информационные системы / К.Н. Мезенцев. - М.: Academia, 2016. - 176 с.
24. Осипов Д. Л. Технологии проектирования баз данных. — М.: ДМК Пресс, 2019. — 498 с.: ил.



25. Попова-Коварцева, Дарья Александровна Основы проектирования баз данных: учеб. пособие / Д.А. Попова-Коварцева, Е.В. Сопченко. – Самара: Изд-во Самарского университета, 2019. – 112 с.: ил.
26. Райордан Р. Основы реляционных баз данных/Пер, с англ. — М.: Издательско-торговый дом «Русская Редакция», 2001. — 384 с.: ил.
27. Ракитов А.И. Системный анализ и аналитические исследования: руководство для профессиональных аналитиков / А.И. Ракитов, Д.А. Бондяев, И.Б. Романов, СВ. Егеров, А.Ю. Щербаков; [ОТВ. ред. А.И. Ракитов]. - М., 2009. - 448 с
28. Редько В.Н. Базы данных и информационные системы / В.Н. Редько, И.А. Басараб. - М.: Знание, 2016. - 667 с.
29. Сергеев С. Ф., Падерно П. И., Назаренко Н. А. Введение в проектирование интеллектуальных интерфейсов: Учебное пособие. - СПб: СПбГУ ИТМО, 2011. - 108 с.
30. Сергеев А. А. Бизнес-планирование: учебник и практикум для среднего профессионального образования / А. А. Сергеев. — 4-е изд., испр. и доп. — Москва: Издательство Юрайт, 2022. — 484 с. — (Профессиональное образование). — ISBN 978-5-534-14036-1. — Текст: электронный // Образовательная платформа Юрайт [сайт]. — URL: <https://urait.ru/bcode/495183> (дата обращения: 10.03.2022).
31. Сибилев В.Д. Модели и проектирование баз данных [Электронный ресурс] URL: [https://www.studmed.ru/sibilev-vd-modeli-i-proektirovanie-baz-dannyh\\_dbf86c7f1ce.html](https://www.studmed.ru/sibilev-vd-modeli-i-proektirovanie-baz-dannyh_dbf86c7f1ce.html) (дата обращения: 12.03.2022).
32. Федорова Г. Н. Информационные системы: учебник для студ. учреждений сред. проф. образования / Г. Н. Федорова. — 3-е изд., стер. — М.: Издательский центр «Академия», 2013. — 208 с.
33. Фуфаев Э. В. Базы данных: учеб. пособие для студ. учреждений сред, проф. образования / Э. В.Фуфаев, Д. Э.Фуфаев. — 7-е изд., стер. — М.: Издательский центр «Академия», 2012. — 320 с.

**Программный продукт «Автоматизированная информационная система  
«Контроль состояния здоровья студентов ГАПОУ СО «Камышловский  
педагогический колледж»**

## Приложения

### Приложение 1

#### Глоссарий сущностей

Наименование сущности	Описание сущности
USER_ROLE (Роль пользователя)	Роль пользователя в приложении, определяет к каким данным и функциям будет доступ
USER (Пользователь)	Лицо, которое будет пользоваться программой
CHECK_VACCINE (Осмотры перед прививками и после прививок)	Документ “Осмотры перед прививками и после прививок”
DATA_MEDICAL_CHECKUPS (Данные медицинских профилактических осмотров)	Документ “Данные медицинских профилактических осмотров”
GROUP (Группа)	Группа студентов, которые проходят обучения в колледже
MEDICAL_ORGANIZATION (Медицинская организация)	Медицинская организация - юридическое лицо или индивидуальный предприниматель, осуществляющий медицинскую деятельность и имеющий на это соответствующую лицензию.
OPERATIONS (Операции)	Операции студента за период обучения в колледже
PASSES_FOR_ILLNESS (Пропуски по болезни)	Пропуски занятий по болезни
PREVENTIVE_VACCINE (Профилактические прививки)	Профилактические прививки студента
PASS_CONROLLER (Ответственный за пропуски)	Студент ответственный за пропуски занятий в группе
SANITARY_SPA_TREATMENT (Санитарно-курортное лечение)	Санитарно-курортное лечение студента
HOSPITAL_TREATMENT (Стационарное лечение)	Стационарное лечение студента
LOGIN_PASSWORD (Логины и пароли)	Таблица содержит логины и пароли пользователей
SURVEYS (Обследования)	Обследования студента за период обучения в колледже
PAST_INFECTIOUS_DIASES (Перенесенные инфекционные заболевания)	Перенесенные студентом инфекционные заболевания
TRAUMA (Травмы)	Травмы студента за период обучения в колледже

## Приложение 2

### Глоссарий атрибутов

Сущность	Атрибут	Тип	Описание
ID_USER_ROLE	USER_ROLE (Роль пользователя)	int	Идентификационный номер роли пользователя
NameRole	USER_ROLE (Роль пользователя)	nvarchar(50)	Наименование роли
ID_USER	USER (Пользователь)	int	Идентификационный номер пользователя
ID_USER_ROLE	USER (Пользователь)	int	Роль пользователя в приложении
Surname	USER (Пользователь)	nvarchar(50)	Фамилия
Name	USER (Пользователь)	nvarchar(50)	Имя
Patronymic	USER (Пользователь)	nvarchar(50)	Отчество
DateBirth	USER (Пользователь)	date	Дата рождения
HomeAddress	USER (Пользователь)	nvarchar(50)	Домашний адрес
Hostel	USER (Пользователь)	nvarchar(50)	Проживает ли пользователь в общежитие
ID_GROUP	USER (Пользователь)	int	Идентификационный номер группы
DateEnrollment	USER (Пользователь)	date	Дата поступления
Color	USER (Пользователь)	nvarchar(50)	Цвет пользователя
ImageName	USER (Пользователь)	nvarchar(50)	Фото пользователя
ID_CHECK_VACCINE	CHECK_VACCINE (Осмотры перед прививками и после прививок)	int	Идентификационный номер документа “Осмотры перед прививками и после прививок”
ID_USER	CHECK_VACCINE (Осмотры перед прививками и после прививок)	int	Идентификационный номер пользователя
Date	CHECK_VACCINE (Осмотры перед прививками и после прививок)	date	Дата

Сущность	Атрибут	Тип	Описание
Body_temp	CHECK_VACCINE (Осмотры перед прививками и после прививок)	int	Температура тела
ResultInspeсion	CHECK_VACCINE (Осмотры перед прививками и после прививок)	nvarchar(50)	Результат осмотра
Vaccine	CHECK_VACCINE (Осмотры перед прививками и после прививок)	nvarchar(50)	Прививка
Permission	CHECK_VACCINE (Осмотры перед прививками и после прививок)	nvarchar(50)	Разрешена или не разрешена
StateAfterThirtyMin	CHECK_VACCINE (Осмотры перед прививками и после прививок)	int	Состояние через 30 мин
ID_DATA_MEDICAL_CHECKUPS	DATA_MEDICAL_CHECKUPS (Данные медицинских профилактических осмотров)	int	Идентификационный номер документа “Данные медицинских профилактических осмотров”
ID_USER	DATA_MEDICAL_CHECKUPS (Данные медицинских профилактических осмотров)	int	Идентификационный номер пользователя
ID_MEDICAL_ORGANIZATION	DATA_MEDICAL_CHECKUPS (Данные медицинских профилактических осмотров)	int	Идентификационный номер медицинской организации
Year	DATA_MEDICAL_CHECKUPS (Данные медицинских профилактических осмотров)	int	Год
Survey	DATA_MEDICAL_CHECKUPS (Данные медицинских профилактических осмотров)	nvarchar(50)	Обследование

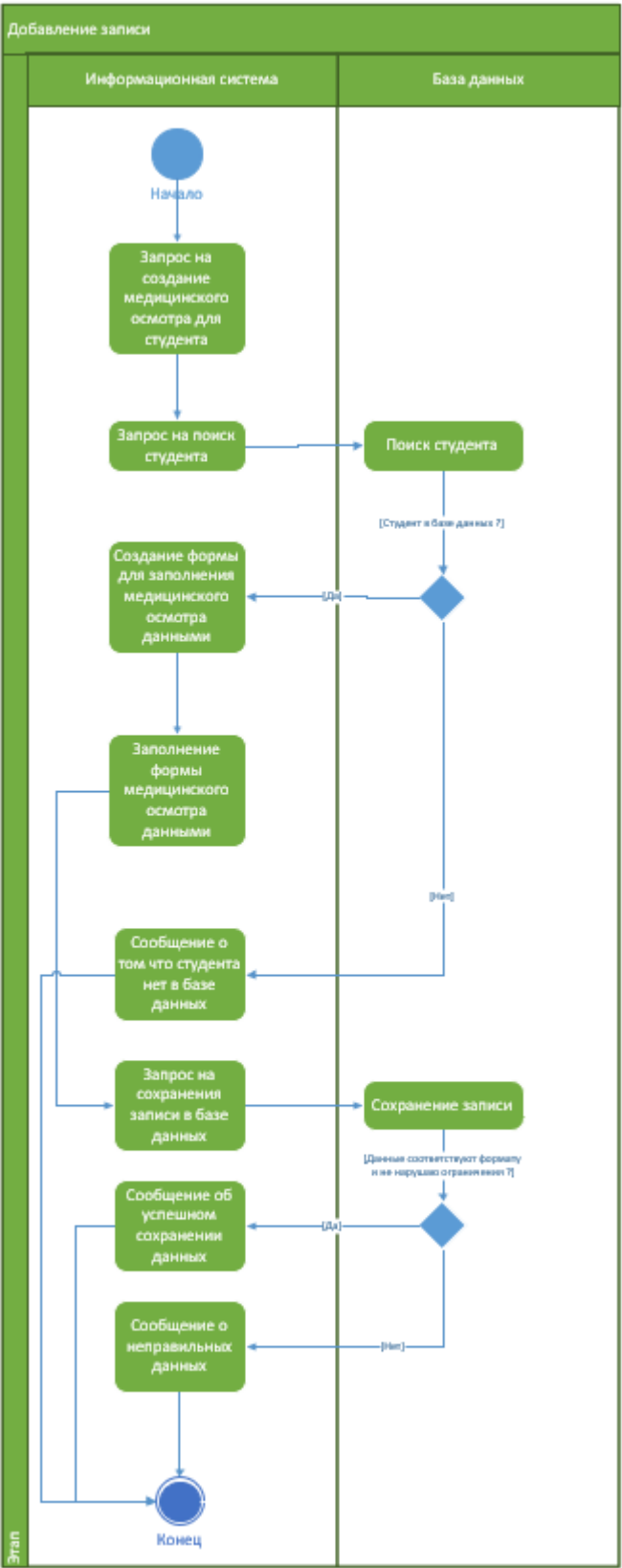
Сущность	Атрибут	Тип	Описание
Survey_result	DATA_MEDICAL_CHECKUPS (Данные медицинских профилактических осмотров)	nvarchar(50)	Результат обследования
ID_GROUP	GROUP (Группа)	int	Идентификационный номер группы
Name_group	GROUP (Группа)	nvarchar(50)	Имя группы
Number_of_persons	GROUP (Группа)	int	Кол-во человек
ID_MEDICAL_ORGANIZATION	MEDICAL_ORGANIZATION (Медицинская организация)	int	Идентификационный номер медицинской организации
Name_medical_organization	MEDICAL_ORGANIZATION (Медицинская организация)	nvarchar(50)	Название медицинской организации
Adres	MEDICAL_ORGANIZATION (Медицинская организация)	nvarchar(50)	Адрес
ID_OPERATIONS	OPERATIONS (Операции)	int	Идентификационный номер документа “Операции”
ID_USER	OPERATIONS (Операции)	int	Идентификационный номер пользователя
Operations	OPERATIONS (Операции)	nvarchar(50)	Операция
ID_MEDICAL_ORGANIZATION	OPERATIONS (Операции)	int	Идентификационный номер медицинской организации
Date	OPERATIONS (Операции)	date	Дата
ID_PREVENTIVE_VACCINE	PASSES_FOR_ILLNESS (Пропуски по болезни)	int	Идентификационный номер документа “Пропуски по болезни”
ID_USER	PASSES_FOR_ILLNESS (Пропуски по болезни)	int	Идентификационный номер пользователя
DateFrom	PASSES_FOR_ILLNESS (Пропуски по болезни)	date	Дата начала

Сущность	Атрибут	Тип	Описание
	по болезни)		
DateTo	PASSES_FOR_ILLNESS (Пропуски по болезни)	date	Дата конца
Diagnosis	PASSES_FOR_ILLNESS (Пропуски по болезни)	nvarchar(50)	Диагноз
ID_PREVENTIVE_VACCINE	PREVENTIVE_VACCINE (Профилактические прививки)	int	Идентификационный номер документа “Профилактические прививки”
ID_USER	PREVENTIVE_VACCINE (Профилактические прививки)	int	Идентификационный номер пользователя
ID_MEDICAL_ORGANIZATION	PREVENTIVE_VACCINE (Профилактические прививки)	int	Идентификационный номер медицинской организации
Vaccine	PREVENTIVE_VACCINE (Профилактические прививки)	nvarchar(50)	Прививка
Date	PREVENTIVE_VACCINE (Профилактические прививки)	date	Дата, когда поставили прививку
ID_PASS_CONTROLLER	PASS_CONTROLLER (Ответственный за пропуски)	int	Идентификационный номер ответственного за пропуски
ID_USER	PASS_CONTROLLER (Ответственный за пропуски)	int	Идентификационный номер пользователя
ID_SANITARY_SPA_TREATMENT	SANITARY_SPA_TREATMENT (Санитарно-курортное лечение)	int	Идентификационный номер документа “Санитарно курортного лечения”
ID_USER	SANITARY_SPA_TREATMENT (Санитарно-курортное лечение)	int	Идентификационный номер пользователя
Description	SANITARY_SPA_TREATMENT (Санитарно-курортное лечение)	nvarchar(50)	Описание
ID_STATS_TREATMENT	HOSPITAL_TREATMENT (Стационарное	int	Идентификационный номер пользователя

Сущность	Атрибут	Тип	Описание
	лечение)		
ID_USER	HOSPITAL_TREATMENT (Стационарное лечение)	int	Идентификационный номер пользователя
Description	HOSPITAL_TREATMENT (Стационарное лечение)	nvarchar(50)	Описание
ID_SURVEYS	SURVEYS (Обследования)	int	Идентификационный номер обследования
ID_USER	SURVEYS (Обследования)	int	Идентификационный номер пользователя
Year	SURVEYS (Обследования)	date	Год
Survey	SURVEYS (Обследования)	nvarchar(50)	Обследование
Survey_result	SURVEYS (Обследования)	nvarchar(50)	Результат обследования
ID_MEDICAL_ORGANIZATION	SURVEYS (Обследования)	int	Идентификационный номер медицинской организации
ID_PAST_INFECTIONOUS_DISEASES	PAST_INFECTIONOUS_DISEASES (Перенесенные инфекционные заболевания)	int	Идентификационный номер документа “Перенесенные инфекционные заболевания”
ID_USER	PAST_INFECTIONOUS_DISEASES (Перенесенные инфекционные заболевания)	int	Идентификационный номер пользователя
Date	PAST_INFECTIONOUS_DISEASES (Перенесенные инфекционные заболевания)	date	Дата
Description	PAST_INFECTIONOUS_DISEASES (Перенесенные инфекционные заболевания)	nvarchar(50)	Заболевание
ID_TRAUMA	TRAUMA (Травмы)	int	Идентификационный номер документа “Травмы”
ID_USER	TRAUMA (Травмы)	int	Идентификационный номер пользователя
Description	TRAUMA (Травмы)	nvarchar(50)	Описание

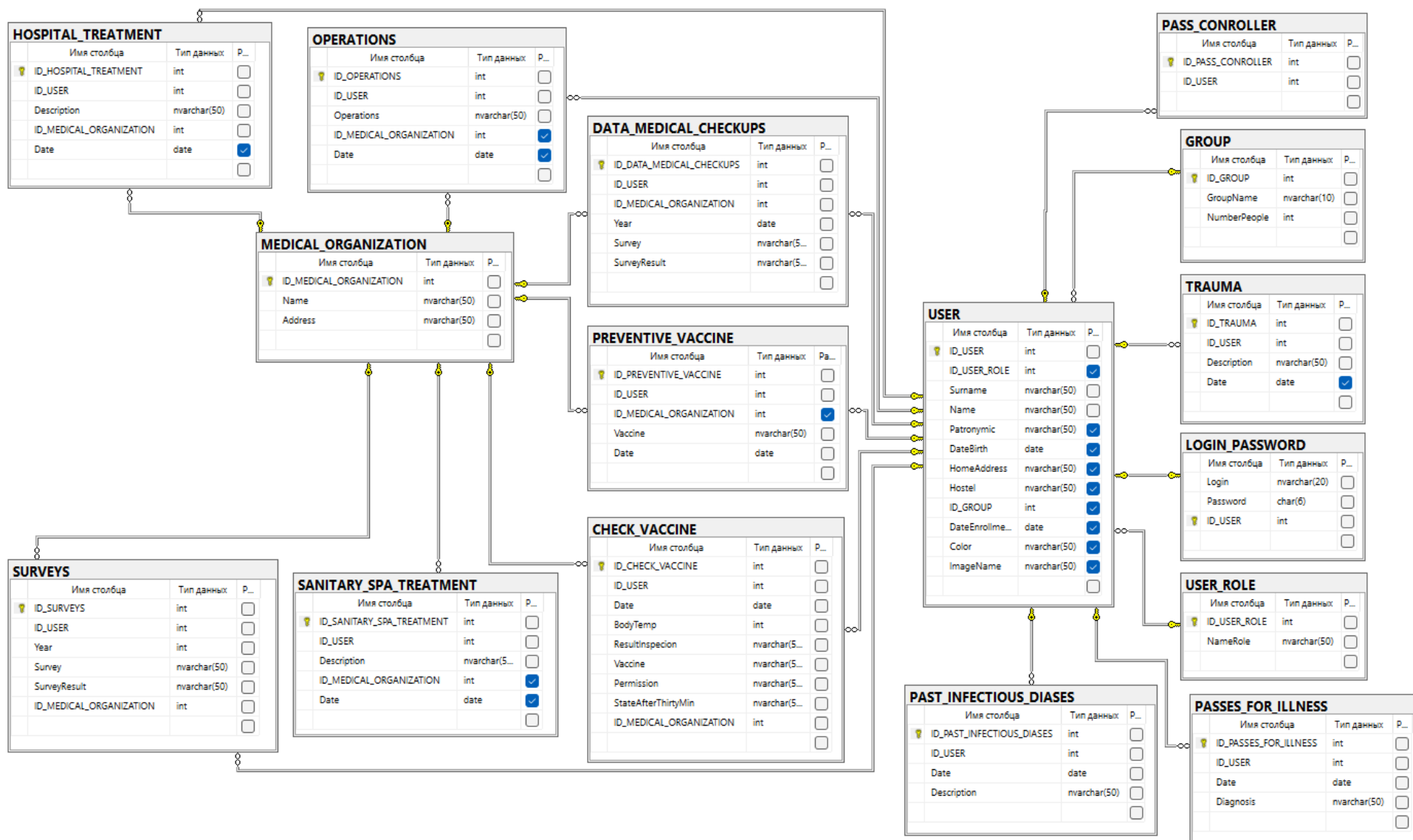


Диаграмма деятельности «Добавление записи»



## Приложение 4

### Логическая модель ФА-уровня



# Физическая модель автоматизированной информационной системы «Контроль состояния здоровья студентов

## ГАПОУ СО «Камышловский педагогический колледж»

