

Министерство науки и высшего образования Российской Федерации
Федеральное государственное автономное образовательное учреждение
высшего образования
«Южно-Уральский государственный университет
(национальный исследовательский университет)»
Высшая школа электроники и компьютерных наук
Кафедра «Информационно-аналитическое обеспечение управления в социальных
и экономических системах»

РАБОТА ПРОВЕРЕНА

Рецензент, руководитель
регионального центра
ЗАО "ПФ"СКБ Контур"

_____ И.М. Гафаров
_____ 2019 г.

ДОПУСТИТЬ К ЗАЩИТЕ

Заведующий кафедрой,
д.т.н., профессор

_____ О.В. Логиновский
_____ 2019 г.

Разработка модуля кадрового учета
информационной системы детской музыкальной школы

Пояснительная записка к выпускной квалификационной работе
ЮУрГУ – 09.04.01.2019.478 ПЗ ВКР

Руководитель ВКР,
д.т.н., профессор

_____ О.В. Логиновский
_____ 2019 г.

Автор ВКР
студент группы КЕ-221

_____ А.И. Дерябина
_____ 2019 г.

Нормоконтролер,
к.т.н., доцент

_____ А.М. Коровин
_____ 2019 г.

Челябинск 2019

АННОТАЦИЯ

Дерябина А.И. Разработка модуля кадрового учета информационной системы детской музыкальной школы. –Челябинск: ЮУрГУ, ВШЭКН, 2019, 95с., 36 илл., 6 табл., библиогр. список –22 наим. Ключевые слова: Информационная система, модуль, С#, база данных.

В рамках выпускной квалификационной работы проведен обзор информационных систем для детских учреждений дополнительного образования, документов и отчетов учреждения дополнительного образования, а так же процессов их подготовки и формирования. Выполнен обзор аналогичных программных решений в сфере образования и выявлены их достоинства и недостатки. Благодаря проведенному анализу сформулированы требования к автоматизированной системе, а также задачи, которые необходимо решить для достижения поставленной цели.

На основе проведенного анализа требований, предъявляемых к разрабатываемой автоматизированной системе, была спроектирована и реализована база данных, разработан алгоритм работы программного обеспечения для управления спроектированной базы данных, выбраны средства реализации автоматизированной системы, реализовано программное обеспечение и составлена необходимая программная документация.

СОДЕРЖАНИЕ

ВВЕДЕНИЕ.....	4
1. ХАРАКТЕРИСТИКА ПРЕДМЕТНОЙ ОБЛАСТИ	6
1.1 Общая характеристика системы детского дополнительного образования.....	6
1.2 Характеристика МБУДО "Детская музыкальная школа № 2"	6
1.3 Обзор документов и отчетов учреждения дополнительного образования.....	7
2. АНАЛИЗ ИНФОРМАЦИОННЫХ СИСТЕМ ДЛЯ УЧРЕЖДЕНИЙ ДЕТСКОГО ДОПОЛНИТЕЛЬНОГО ОБРАЗОВАНИЯ.....	13
2.1 Информационные потоки учреждений дополнительного образования	13
2.2 Сравнение существующих информационных систем для учреждений детского дополнительного образования.....	15
2.3 Требования, предъявляемые к разрабатываемому модулю кадрового учета	23
3. РАЗРАБОТКА МОДУЛЯ КАДРОВОГО УЧЕТА	26
3.1 Выбор средств разработки.....	26
3.2 Проектирование схемы базы данных.....	28
3.3 Проектирование программного обеспечения.....	39
3.4 Разработка программного обеспечения.....	44
3.5 Описание модуля кадрового учета.....	65
ЗАКЛЮЧЕНИЕ.....	79
БИБЛИОГРАФИЧЕСКИЙ СПИСОК	80
ПРИЛОЖЕНИЕ А.....	84

ВВЕДЕНИЕ

Внедрение информационных технологий хранения, обработки, передачи информации обладает весомой значимостью для развития экономики и её возможностей, и является современным направлением мирового развития последних нескольких лет.

Внедрение информационных систем в повседневную деятельность образовательного учреждения позволит:

1. уменьшить количество бумажных данных и перейти на электронный вид управления;
2. стандартизировать процессы управления;
3. повысить эффективность управленческих решений на основе использования оперативной, достоверной и актуальной информации.

Сфера деятельности любого учреждения дополнительного образования охватывает целую совокупность процессов, управление которыми, является главной функцией образования в целом. К таким процессам относится и процесс управления кадрами.

Во многих учреждениях дополнительного образования кадровый учет вводится с помощью электронных таблиц, а формирование отчетов о состоянии кадров производится вручную. Так как статистические отчеты необходимо составлять периодически, то данный процесс занимает много времени, а так же способ заполнения их вручную не отличается точностью. Поэтому автоматизация кадрового учета является актуальной задачей.

Автоматизация процесса управления кадрами помогает более рационально распределять трудовые ресурсы и управлять денежными потоками, но и являться источником, из которого можно получить сведения как корпоративного, так и индивидуального характера. Информационные системы могут вести учет работников, регистрировать и издавать приказы, следить за предоставлением отпусков, получать разнообразную аналитическую информацию и решать многие другие задачи.

Заказчиком разрабатываемой автоматизированной системы является муниципальное бюджетное образовательное учреждение дополнительного образования детей «Детская музыкальная школа № 2». Детская музыкальная школа №2 основана в 1956 г. и является одной из крупнейших школ горнозаводского края. На сегодняшний день главный акцент в концепции развития школы делается на инновационную деятельность, проекты музыкальных программ и постановок.

Целью данной выпускной квалификационной работы является повышение эффективности управления кадрами в учреждении дополнительного образования с помощью автоматизации кадрового документооборота.

Объектом выпускной квалификационной работы является программное обеспечение для модуля кадрового учета детского дополнительного образовательного учреждения.

Предметом выпускной квалификационной работы является кадровый документооборот муниципального бюджетного образовательного учреждения дополнительного образования детей на примере «Детской музыкальной школы № 2».

Задачи выпускной квалификационной работы:

- провести анализ документов и отчетов, с которыми работает муниципальное бюджетное образовательное учреждение дополнительного образования детей «Детская музыкальная школа № 2»;
- проанализировать существующие программные решения для учета кадров в образовательных учреждениях;
- спроектировать базу данных, удовлетворяющую требованиям к разрабатываемому программному продукту;
- разработать алгоритм работы программного обеспечения;
- выбрать средства реализации данного программного продукта;
- реализовать программное обеспечение.

1 ХАРАКТЕРИСТИКА ПРЕДМЕТНОЙ ОБЛАСТИ

1.1 Общая характеристика системы детского дополнительного образования

Детское дополнительное образование – это особая подсистема общего образования, обеспечивающая развитие интересов и способностей личности, ее индивидуальный образовательный путь на основе свободного выбора содержательной деятельности, которая не ограничивается рамками образовательных стандартов и формами традиционной внеурочной и внешкольной работы.

Дополнительное образование позволяет человеку приобрести устойчивую потребность в познании и творчестве, максимально реализовать себя, самоопределиваться предметно, социально, профессионально, личностно.

К муниципальным образовательным учреждениям дополнительного образования детей относятся:

Центры дополнительного образования различной направленности (научно-техническое, оздоровительное экологическое, искусственное, туристическое и т.д.)

- Дворцы детского и юношеского творчества;
- Дома детского творчества;
- Станции юных натуралистов;
- Детские школы искусств;
- Детско-юношеские спортивные школы;

1.2 Характеристика МБУДО "Детская музыкальная школа № 2"

Муниципальное бюджетное учреждение дополнительного образования «Детская музыкальная школа № 2» Златоустовского городского округа осуществляет деятельность в соответствии с действующим законодательством, уставом учебного заведения, законом Российской Федерации "Об образовании". МБУДО "Дет-

ская музыкальная школа № 2" ЗГО оказывает образовательные услуги несовершеннолетним гражданам, достигшим возраста пяти лет и до достижения ими возраста восемнадцати лет, по предоставлению дополнительного образования.

Детская музыкальная школа № 2 осуществляет свою деятельность на основании свидетельства о Государственной аккредитации, лицензию на право ведения образовательной деятельности.

В школе реализуются образовательные программы в области искусств по следующим направлениям:

- Фортепиано;
- Хоровое пение;
- Струнные инструменты;
- Духовые инструменты;
- Народные инструменты;
- Эстетическое образование.

Музыкальная школа имеет в своем составе административно – управленческий отдел: директор, заместитель директора по учебно-воспитательной работе, заместитель директора по научно-методической работе, заместитель по административно-хозяйственной части, главный бухгалтер, секретариат (делопроизводитель); методический отдел (методист), бухгалтерия (бухгалтеры); педагогический отдел (педагоги дополнительного образования и педагоги-организаторы), технический отдел (сторожа, технички, вахтеры, слесарь, плотник и автослесарь).

1.3 Обзор кадрового документооборота и отчетов учреждения дополнительного образования

Для того чтобы провести анализ отечественных и передовых зарубежных технологий и решений, необходимо изучить документы и отчеты предприятия до начала этапа разработки автоматизированной системы управления.

В качестве документа, хранящего информацию о сотрудниках учреждения, используется файл формата электронной таблицы .xls. Его структура представлена на рисунке 1.1.

№ п/п	ФИО	Должность	Дата рождения	Категория	Образование	Стаж (лет)	Филиал школы

Рисунок 1.1– Структура таблицы с информацией о сотрудниках

Согласно приказа Росстата от 15.07.2011 № 324 (ред. от 28.02.2013) «Об утверждении статистического инструментария для организации Минкультуры России федерального статистического наблюдения за деятельностью учреждений культуры» [1] форма федерального статистического наблюдения N 1-ДМШ «Сведения о детской музыкальной, художественной, хореографической школе и школе искусств» [1] форма федерального статистического наблюдения N 1-ДМШ «Сведения о детской музыкальной, художественной, хореографической школе и школе искусств» заполняется всеми детскими музыкальными, художественными, хореографическими школами и школами искусств. Данные о работе филиалов включаются в форму, представляемую головной музыкальной, художественной, хореографической школой и школой искусств.

Данная форма отчета представляется юридическими лицами (детские музыкальные, художественные, хореографические школы и школы искусств) в орган самоуправления, осуществляющий управление в сфере культуры, и в орган исполнительной власти субъекта Российской Федерации, осуществляющий управление в сфере культуры.

Данная форма отчета состоит из следующих разделов:

- материально – техническая база;
- численность учащихся на начало учебного года;
- численность работников, состав преподавателей и концертмейстеров;
- поступление и использование финансовых средств;
- размер финансовых средств, поступающих от родительских взносов (добровольные пожертвования или средства за оплату обучения).

Информация о сотрудниках отображается только в разделе «Численность работников, состав преподавателей и концертмейстеров», поэтому рассмотрим подробнее данную форму раздела, которая представлена на рисунке 1.2. Все графы данного раздела заполняются на начало учебного года.

N строки	Наименование должностей	Всего работников на конец года, человек	Из них (из гр. 3)		Из числа штатных работников (из гр. 5), работающих на условиях штатного совместительства	Из общей численности штатных работников (из гр. 5) имеют образование						Из общей численности штатных работников (из гр. 5) имеют стаж работы		
			имеют подготовку по специальности по использованию ИКТ (сумма гр. 7, 9, 11)	штатных работников (сумма гр. 7, 9, 11)		высшее профессиональное	из них по профилю преподаваемого предмета	среднее профессиональное	из них по профилю преподаваемого предмета	другое	до 5 лет	от 6 до 10 лет	от 11 до 25 лет	свыше 25 лет
1	2	3	4	5	6	7	8	9	10	11	12	13	14	15
41	Всего						X		X					
	Из них:													
42	руководитель						X		X					
43	заместитель руководителя						X		X					
44	преподаватель													
45	концертмейстер													
46	методист						X		X					

Рисунок 1.2 – Форма «Численность работников, состав преподавателей и концертмейстеров».

В графу 3 вносятся сведения об общей численности преподавателей и концертмейстеров, административно-управленческого, технического и обслуживающего персонала, как штатных, так и нештатных. Приводятся сведения о фактической численности работников, работающих на условиях полной и частичной занятости, а не по штатному расписанию. Если штатный работник совмещает должности, то он учитывается один раз по основной должности; если штатный работник помимо основной должности работает по договору, то он учитывается дважды, трижды (в зависимости от числа заключенных договоров).

В графе 4 показывается численность сотрудников, которые имеют подготовку по использованию информационно-коммуникационных технологий (ИКТ), из общего числа сотрудников. Указывается численность сотрудников учреждения культуры, прошедших обучение в своем учреждении, или окончивших курсы переподготовки в области использования ИКТ, или имеющих специальное образование в области ИКТ.

В графе 5 указывается численность всех штатных работников учреждения, в том числе работающих на условиях штатного совместительства.

В графе 6 (из графы 5) показывается численность работников, работающих на условиях штатного совместительства.

В графе 7 (из графы 5) показывается численность работников, имеющих высшее профессиональное образование.

В графе 8 (из графы 7) указывается численность работников, имеющих высшее профессиональное образование по профилю преподаваемого предмета.

В графе 9 (из графы 5) приводится численность работников, имеющих среднее профессиональное образование (строки 44, 45).

В графе 10 (из графы 9) указывается численность работников, имеющих среднее профессиональное образование по профилю преподаваемого предмета (строки 44, 45).

В графе 11 (из графы 5) показывается численность работников, имеющих другое образование.

2. АНАЛИЗ ИНФОРМАЦИОННЫХ СИСТЕМ ДЛЯ УЧРЕЖДЕНИЙ ДЕТСКОГО ДОПОЛНИТЕЛЬНОГО ОБРАЗОВАНИЯ

2.1 Информационные потоки учреждений дополнительного образования

В современных быстро меняющихся условиях функционирования УДОД без использования возможностей информационных систем и технологий практически невозможно эффективно решать целый ряд важных задач образовательного учреждения. К таким задачам относятся: обеспечение руководства УДОД исчерпывающей и достоверной информацией для обоснования и принятия управленческих решений; интеграция УДОД в общее информационное пространство общеобразовательных учреждений; обеспечение органов управления образованием инструментарием мониторинга работы УДОД в режиме реального времени; сокращение трудозатрат на подготовку различного рода отчетности [12].

Информационная система (ИС) - это программно-аппаратный комплекс, предназначенный для автоматизированного сбора, хранения, обработки и выдачи информации. Обычно ИС имеют дело с большими объемами информации, которая имеет достаточно сложную структуру [13].

Наибольшее применение информационные системы в образовательных учреждениях находят при автоматизации следующих задач: — управление учебным процессом (формирование учебных планов, построение расписания, мониторинг результатов обучения); — финансовое планирование и бухгалтерский учет; — учет состояния материально-технической базы; — управление персоналом (штатное расписание, персональные данные, движение персонала, повышение квалификации); — автоматизация деятельности библиотек; — документооборот (формирование приказов, контроль исполнения); — подготовка оперативной и внешней отчетности [17].

Внедрение информационных систем в повседневную деятельность образовательного учреждения позволит:

1. уменьшить количество бумажных данных и перейти на электронный вид управления;

2. стандартизировать процессы управления;

3. повысить эффективность управленческих решений на основе использования оперативной, достоверной и актуальной информации [19].

Деятельность УДОД состоит из административного управления и управления учебным процессом. К функциям административного управления относятся: осуществление приема на работу кадров, распределение должностных обязанностей, утверждение штатного расписания, определение оклада, надбавок и доплат, распоряжение материальным имуществом учреждения. К функциям педагогического управления относятся: планирование и организация образовательного процесса, составление образовательных программ.

В документообороте музыкальной школы участвуют следующие виды документов:

- Заявление от учеников и поступление;
- Расписание учебных занятий;
- Оплата образовательных услуг.

Исходя из выше сказанного, к информационной системе детского учреждения дополнительного образования будут предъявляться следующие требования:

- хранение информации;
- учет информации;
- анализ информации.

Информационные потоки любого учреждения отражены в его документообороте и распределяются по следующим направлениям:

- Сведения о клиентах: ФИО ребенка и попечителя, возраст и пол ребенка, адрес его проживания и контактный телефон;
- Сведения о специалистах: ФИО, дата рождения, контактный телефон, адрес проживания, номер паспорта, должность и дата приема на работу;
- Сведения о должностях специалистов: наименование;
- Сведения об услугах: наименование и количество часов.

Информационная система детского учреждения дополнительного образования должна иметь следующие модули.

1. Общее делопроизводство - позволяет автоматизировать работу с различными документами и осуществлять контроль исполнения поручений руководителя и документов.

2. Учет воспитанников - призван помочь заведующему в учете воспитанников учреждения и при необходимости помочь в формировании очереди на поступление в детский сад.

3. Кадровый учет - позволяет осуществлять учет сотрудников организации с момента их приема в организацию до момента увольнения, отслеживать их перемещения по карьерной лестнице, квалификацию сотрудников, т.е. решать основные задачи кадрового учета организации.

4. Учет материальных ценностей - предназначен для учета всех материальных ценностей организации (в том числе и не находящихся на балансовом учете). Она позволяет собирать информацию о том, где эти ценности находятся и кто назначен ответственным за их хранение.

5. Учет методических материалов - предназначен для учета существующих методических материалов организации, учета выдачи материалов на руки сотрудникам и последующего возврата. Данная подсистема позволяет работать с отдельными материалами (книгами, журналами и т.д.) в электронной форме и позволяет существенно упростить специалистам-педагогам подбор материалов к занятиям.

2.2 Сравнение существующих информационных систем для учреждений детского дополнительного образования

На данный момент существует множество программных продуктов для автоматизированного управления образовательным учреждением. Однако, программного обеспечения для автоматизации работы отдела кадров музыкальной

школы в настоящее время не существует. Поэтому рассмотрим программные решения для образовательных учреждений.

Автоматизированная информационно-аналитическая система «Управление образовательным учреждением (АРМ Директор)». Специализированная система, предназначенная для унификации внутришкольного делопроизводства, автоматизации процессов управления школой, планирования и контроля качества учебного процесса. Разработчиком данной системы является российская фирма «Аверс».

Условно программу можно разбить на несколько функциональных подсистем, с которыми работают различные пользователи:

1) подсистема делопроизводства позволяет:

- вести личные дела сотрудников учреждения и школьников;
- формировать адресную и алфавитную книги;
- формировать стандартные отчеты ОШ-1, ОШ-5, ОШ-9, паспорт школы, наполняемости классов, о составе учащихся по полу и по годам;
- выводить на печать карточку учащегося и личное дело сотрудника (Форма Т-2);
- создавать проекты приказов по учебному заведению из предложенных шаблонов, формировать циклограмму приказов учебного заведения, вести электронные книги приказов по учащимся и персоналу школы [2].

2) подсистема планирования:

- рассчитывает базисный и учебный планы, сетку часов на основе сформированной структуры школы, списков классов, базового и школьного компонентов;
- позволяет распределять основную и дополнительную нагрузку преподавателей;
- предоставляет данные для формирования расписания занятий при совместной работе с программами составления расписаний наиболее известных и востребованных фирм.

3) подсистема успеваемости учитывает успеваемость учащихся:

- в виде: оценок, по принятой в учреждении шкале;
- в виде обобщенных показателей, рассчитываемых программой на основе показателей, заданных пользователем.

4) подсистема аналитики позволяет проводить анализ состояния учебного процесса по критериям:

- степени обученности;
- качества знаний;
- проценту успеваемости учащихся;
- среднему баллу [2].

Стоимость данного программного продукта составляет от 38 000 до 59 000 рублей в зависимости от заказанных подсистем.

К недостаткам данной системы можно отнести:

- невозможность формирования списка сотрудников в соответствии с распределением их на филиалы учреждения;
- невозможность формирования отчетов формы 1-ДМШ и о кадровом обеспечении;
- высокая стоимость;
- излишняя модульность, которая загромождает основной рабочий процесс пользователей сложностью.

Автоматизированная система управления вузом (АСУ ВУЗ) «UniversysWS 3.5» - это программная платформа для управления образовательными организациями различных уровней. Данная система разработана компанией «Гисофт».

АСУ ВУЗ «UniversysWS 3.5» представляет собой программно-аппаратный комплекс, состоящий из сервера, базы данных (БД) и программного обеспечения (ПО) системы.

ПО АСУ ВУЗ «UniversysWS 3.5» обеспечивает работу виртуальных модулей системы [3]:

- модуль ректорат;

- модуль деканат;
- модуль отдел кадров;
- модуль приемная комиссия;
- модуль бухгалтерия;
- модуль библиотека;
- модуль личный кабинет преподавателя.

Некоторые основные функциональные возможности АСУ ВУЗ «UniversysWS 3.5»:

- формирование структуры образовательной деятельности и соответствующих учебных планов;
- оформление юридических взаимоотношений со студентами или их доверенными лицами;
- учет финансовых поступлений (оплат за обучение), оформление и печать бухгалтерских документов об оплате;
- учет студентов, структурного распределения студентов в организационной, научной и других структурах образовательного учреждения;
- ведение учета и планирование (составление расписаний) учебных мероприятий (лекций, семинаров, практических занятий), их содержания и посещаемости;
- ведение контроля учебного процесса в режиме реального времени («online»);
- генерация и печать всех необходимых документов, сопровождающих учебный процесс;
- ведение учета в документообороте, автоматизация процедур создания и согласования документов, архивирование документов, информирование студентов и персонала о действующих распорядительных и иных документах;
- учет персонала по категориям.

Право эксплуатации данного программного продукта осуществляется на основании предоставленной Лицензии. Лицензия предоставляется на срок от 1 до 3

лет. Расходы на Лицензию для одного сотрудника или преподавателя составляют 1200 рублей в год.

К недостаткам данного программного продукта также относятся:

- невозможность формирования отчетов формы 1-ДМШ и о кадровом обеспечении;
- высокая стоимость в случае подписания длительного лицензионного соглашения;
- излишняя функциональность.

Программа "Сотрудники предприятия" предназначена для автоматизации кадровых отделов в организациях. Разработчиком данного программного продукта является фирма «AraXGroup». Программное обеспечение является бесплатным.

Основные возможности программы [4]:

- ведение баз сотрудников сразу по нескольким организациям и предпринимателям в одном экземпляре программы;
- контроль вакансий;
- построение различных выборок по заданным критериям: уволенные работники, работники определенного возраста, дни рождения работников;
- заполнение в автоматическом режиме стандартных форм: личное дело, табель учета рабочего времени, приказы о приеме на работу и увольнении, о предоставлении отпусков и т.д.;
- составление журналов в автоматическом режиме: трудовых договоров, командировочных приказов и всевозможных отчетов по сотрудникам;
- изменение предустановленных шаблонов и создание собственных шаблонов трудового договора, приказов, договора о материальной ответственности и других документов;
- расчет отпусков и трудового стажа;

Программа может работать в сети, количество пользователей программы не лимитируется. Есть разграничение прав доступа.

К недостаткам данного программного продукта относятся:

- отсутствие функции резервного копирования базы данных;
- невозможность формирования списка сотрудников в соответствии с распределением их на филиалы учреждения;
- невозможность формирования отчетов формы 1-ДМШ и о кадровом обеспечении;
- нестабильность работы программы при увеличении количества записей в базе данных.

Программный продукт «1С: Зарплата и кадры образовательного учреждения 8» позволяет автоматизировать ведение учета, распределять денежные средства, направляемые в учреждение образования в соответствии с нормативным подушевым финансированием, и проводить все расчеты по оплате труда в соответствии с новой системой оплаты труда [20]. Разработчиком данного программного продукта является фирма «1С».

«1С: Зарплата и кадры образовательного учреждения 8» позволяет проводить все работы от формирования фонда оплаты труда образовательного учреждения до начисления заработной платы сотрудникам образовательных учреждений в одной информационной базе.

Для расчета и учета заработной платы в программе автоматизирована деятельность бухгалтеров-расчетчиков заработной платы[5]:

- автоматический расчет широкого круга начислений - от оплаты по окладу и различных надбавок до оплаты больничных и отпусков по среднему заработку;
- гибкая настройка используемых начислений и удержаний.

«1С:Зарплата и кадры образовательного учреждения» предоставляет широкие возможности учета рабочего времени. Гибкий механизм позволяет, с одной стороны, описывать различные графики работы, включая «скользящие», и регистрировать только отклонения от обычного режима работы, а с другой – регистрировать только фактические данные учета на основе заполняемых в подразделениях табелей [20].

Данный программный продукт поддерживает ведение штатного расписания учреждений с возможностью указания различных видов тарифных ставок, произвольного количества надбавок, дополнительной информации о штатных единицах. По штатному расписанию формируется вся необходимая отчетность, в том числе представление штатного расписания в виде унифицированной формы.

В программе автоматизировано кадровое делопроизводство, включая заполнение унифицированных печатных форм: оформление трудовых договоров, прием на работу (формы Т-1 и Т-1а), кадровые перемещения работников (формы Т-5 и Т-5а), увольнение из учреждения (формы Т-8 и Т-8а).

На основании кадровых данных строится унифицированная форма Т-2, содержащая в том числе сведения о непрерывном, общем стаже и стаже на надбавку за выслугу лет работников учреждения.

В соответствии с утвержденным графиком отпусков учреждения заполняется унифицированная форма Т-7 и формируются приказы о предоставлении отпуска работникам (формы Т-6 и Т-6а) [20].

Стоимость данного программного продукта составляет 18000 рублей. Программный продукт «1С: Зарплата и кадры образовательного учреждения 8» обеспечивает работу прикладного решения на одном рабочем месте в один момент времени. Для работы в многопользовательском режиме необходимо наличие у пользователей клиентских лицензий «1С:Предприятия 8». Стоимость клиентской лицензии составляет 4500 рублей. Для использования прикладного решения в варианте «клиент-сервер» необходимо наличие у пользователей лицензии на сервер «1С:Предприятие 8.1», стоимость которой составляет 36000 рублей.

К недостаткам данного программного продукта относятся:

- невозможность формирования списка сотрудников в соответствии с распределением их на филиалы учреждения;
- невозможность формирования отчета формы 1-ДМШ;
- излишняя функциональность;
- высокая стоимость программного продукта.

SIS – программное приложение для образовательных учреждений, управляющее данными ученика. Эта система реализуется в относительно небольших организациях. Разработчиком данной системы является компания Custom–DB. Функциональность SIS может быть расширена за счет дополнительных модулей.

Данная система обеспечивает следующие функции[6]:

- обработка запросов от потенциальных студентов;
- автоматическое создание класса и расписания;
- обработка данных об академических успехах;
- запись коммуникаций со студентами;
- поддержание дисциплины записей;
- предоставление статистических отчетов;
- учет и бюджетирование услуг.

Изменения и обновления в этой системе, как правило, существенно влияют на повседневную деятельность каждого работника школы. По этой причине, следует рассмотреть особенности данной системы:

- рабочий процесс. Поскольку программа тесно привязана к школьному документообороту, изменение системы SIS может вызвать изменения в рабочем процессе. Это может оказать существенное влияние на ежедневные операции;

- преобразование данных. Поскольку в большинстве школ требуется сохранение архивных записей о прошлых учениках, то внимание должно быть уделено тому, какая информация будет преобразована, и что будет архивироваться;

- обучение. В новые версии программы включают новые функции, поэтому обучение сотрудников, чтобы использовать новые SIS программы, скорее всего, будет дорогостоящим и трудоемким процессом.

К недостаткам данного программного продукта также относятся:

- несоответствие выходных форм отчетов требуемым отчетам;
- невозможность формирования списка сотрудников в соответствии с распределением их на филиалы учреждения;
- интерфейс программы на английском языке.

2.3 Требования, предъявляемые к разрабатываемому модулю кадрового учета

Проведя анализ существующих программных решений и требований учреждения-заказчика, приходим к выводу, что разрабатываемая программная система должна содержать следующие требования, предъявляемые к:

1) базе данных:

- база данных должна содержать справочники:
 - образовательных учреждений, которые могут окончить сотрудники музыкальной школы;
 - специализаций образовательных учреждений и сотрудников;
 - должностей с указанием группы, к которой относится данная должность (администрация, преподаватель, концертмейстер, методист, технический персонал);
 - категорий, которые может получить сотрудник;
 - видов совместительства;
 - филиалов школ;
- в базе данных должен учитываться общий стаж работника, а также стаж работы по специальности;

2) функциям системы:

- возможность формирования отчетов формы 1-ДМШ о кадровом обеспечении, а также экспорт данных отчетов в электронную таблицу Excel;
- получение выходных данных с минимальными промежутками времени с момента получения запроса системой;
- обеспечение конфиденциальности информации. Так как в системе должны храниться персональные данные о сотрудниках, то доступ к ним должен быть ограничен от внешних воздействий посторонних лиц. Проблема защиты информации от несанкционированного доступа является одной из важнейших при построении автоматизированной системы. Основными способами защиты персо-

нальных данных от несанкционированного доступа могут быть представлены следующим образом:

- физическая защита ПК и носителей информации;
- аутентификация пользователей в системе;
- разграничение прав доступа к элементам защищаемой информации;
- регистрация всех обращений к защищаемой информации.

Для разграничения прав доступа в системе должны быть предусмотрены следующие пользователи:

- администратор (обеспечивает корректную работу с базой данных);
- директор (заполняет информацию о сотрудниках, получает отчеты);
- бухгалтер (получает информацию о сотрудниках и отчеты).

Для представления ролей пользователей построим диаграмму вариантов использования, изображенную на рисунке 2.4.



Рисунок 2.4 – Диаграмма вариантов использования

Целью выпускной квалификационной работы является повышение эффективности управления кадрами в учреждении дополнительного образования с помощью автоматизации документооборота отдела кадров.

Для достижения поставленной цели необходимо решить следующие задачи:

- провести анализ документов и отчетов, с которыми работает муниципальное бюджетное образовательное учреждение дополнительного образования детей «Детская музыкальная школа № 2»;
- проанализировать существующие программные решения для учета кадров в образовательных учреждениях;
- спроектировать базу данных, удовлетворяющую требованиям к разрабатываемому программному продукту;
- разработать алгоритм работы программного обеспечения;
- выбрать средства реализации данного программного продукта;
- реализовать программное обеспечение.

Выводы по разделу два

Выполнен обзор аналогичных программных решений в сфере образования и выявлены их достоинства и недостатки. Благодаря проведенному анализу были сформулированы требования к разрабатываемой автоматизированной системе, а также задачи, которые необходимо решить для достижения поставленной цели.

3 РАЗРАБОТКА МОДУЛЯ КАДРОВОГО УЧЕТА

3.1 Выбор средств разработки

C# – объектно-ориентированный язык программирования. Разработан в 1998–2001 годах группой инженеров под руководством Андерса Хейлсберга компанией Microsoft как язык разработки приложений для платформы Microsoft .NET Framework. C# относится к семье языков с C-подобным синтаксисом, из них его синтаксис наиболее близок к C++ и Java [7]. При этом язык является проще и надежнее C++, так как на C# хотя и допускается, но не рекомендуется использование таких опасных свойств C++, как указатели, адресация, разыменовывание, адресная арифметика. Язык поддерживает полиморфизм, перегрузку операторов (в том числе операторов явного и неявного приведения типа), делегаты, атрибуты, события, свойства, обобщённые типы и методы, итераторы, исключения, комментарии в формате XML, «сборку мусора», поддержку совместимости версий[8].

Переняв многое от своих предшественников–языков C++, Pascal, Modula, Smalltalk и в особенности Java – C#, опираясь на практику их использования, исключает некоторые модели, зарекомендовавшие себя как проблематичные при разработке программных систем. В нем собраны все достоинства разных языков. Скорость выполнения приближается к языку Assembler. К преимуществу языка также относится наличие более 300 000 библиотек разных функций, облегчающих разработку программного обеспечения.

Среда разработки Visual Studio.NET предоставляет мощные и удобные средства написания, корректировки, компиляции, отладки и запуска приложений, использующих .NET-совместимые языки. Корпорация Microsoft включила в платформу средства разработки для четырех языков: C#, VB.NET, C++ и J#. [7]

Платформа .NET является открытой средой. Это значит, что компиляторы для нее могут поставляться и сторонними разработчиками. Для обеспечения переносимости компиляторы, входящие в состав платформы, переводят программу

не в машинные коды, а в промежуточный язык (Common IntermediateLanguage, CIL, или просто IL), который не содержит команд, зависящих от языка, операционной системы и типа компьютера. Программа на этом языке выполняется не самостоятельно, а под управлением системы, которая называется общеязыковой средой выполнения (Common Language Runtime, CLR).

Во время работы программы среда CLR следит за тем, чтобы выполнялись только разрешенные операции, осуществляет распределение и очистку памяти и обрабатывает возникающие ошибки. Это многократно повышает безопасность и надежность программ.

Платформа .NET содержит огромную библиотеку стандартных классов, которые можно использовать при программировании на любом языке .NET. Библиотека классов вместе с CLR образуют каркас (framework), то есть основу платформы.

Среда Visual Studio.NET позволяет создавать проекты различных типов. Несколько проектов можно объединить в решение (solution). Это облегчает совместную разработку проектов.

СУБД Firebird– это современная система управления реляционными базами данных (СУРБД), применяемая для разработки сложных приложений на базе технологии «клиент-сервер». Данная СУБД основана на открытом исходном коде СУБД Interbase 6.0.

SQL-сервер Firebird предназначен для хранения и обработки больших объемов данных в условиях одновременной работы с БД множества клиентских приложений. Основная цель – уменьшение риска потери или разрушения данных в случае несанкционированного доступа в многопользовательской и конкурентной среде.

Доступ к базе данных может осуществляться с использованием сетевого соединения (например, по протоколу TCP/IP) в случае, если производится подключение к удаленному серверу. Осуществление доступа к базе данных, расположенной на удаленном сервере, производится на компьютере с установленным

FirebirdClient. Он посредством сети взаимодействует с FirebirdServer, получая доступ к базе данных.

Информация обо всех пользователях баз данных СУБД Firebird хранится в общей базе данных безопасности, который содержит информацию о зарегистрированных пользователях и их паролях.

СУБД Firebird содержит различные утилиты командной строки для администрирования БД. Это `gbak.exe`, `nbackup.exe` (средства резервного копирования и восстановления БД для предотвращения потери данных), `isql.exe` (SQL командной строки), `gsec.exe` (для управления пользователями и их паролями), `gstat` (для сбора статистики по БД) [9].

В качестве преимуществ Firebird можно отметить многоверсионную архитектуру, обеспечивающую параллельную обработку оперативных и аналитических запросов (это возможно потому, что читающие пользователи не блокируют пишущих), компактность (дистрибутив 5Mb), высокую эффективность и мощную языковую поддержку для хранимых процедур и триггеров.

3.2 Проектирование схемы базы данных

Для разработки данной автоматизированной системы необходимо осуществить проектирование и реализацию базы данных, отвечающую третьей нормальной форме.

База данных представляет собой совокупность структурированных данных, хранимых в памяти вычислительной системы и отображающих состояние объектов и их взаимосвязей в рассматриваемой предметной области.

Проектирование баз данных состоит из трёх основных фаз: концептуального, логического и физического моделирования.

Целью фазы концептуального проектирования является создание концептуальной модели базы данных. Для этого производится сбор, анализ и редактирование требований к данным. Данный этап был рассмотрен в пунктах 1.1 и 1.3. В

результате анализа выявленных требований было принято решение о том, что ряд информации будет храниться в виде справочных таблиц, а именно данные о видах:

- филиалов музыкальной школы;
- отделений, которые относятся к одному из филиалов;
- категорий, которые может получить сотрудник;
- совместительств сотрудников;
- отпусков;
- уровней полученного образования;
- специализаций;
- должностей;
- групп должностей;
- статусов удаления записи из таблицы с информацией о сотрудниках.

Организация данных в виде справочных таблиц позволит формировать необходимые отчеты, а также является удобным представлением данных для получения необходимой информации. Также это поможет при заполнении базы данных, так как данные, например, о видах совместительства или о должностях сотрудников, можно будет выбрать из справочника, а не заносить информацию вручную. Это обеспечит целостность данных в проектируемой системе.

Информация о сотрудниках будет разбита на следующие таблицы:

- образование сотрудника;
- полученные категории;
- занимаемые сотрудником должности;
- виды совместительства сотрудника;
- предоставленные сотруднику отпуска.

Во время логического моделирования разрабатывается модель базы данных, не зависящая от конкретной СУБД и учитывающая требования, выявленные на этапе концептуального проектирования. На рисунке 3.1 изображена схема базы данных разрабатываемой системы.

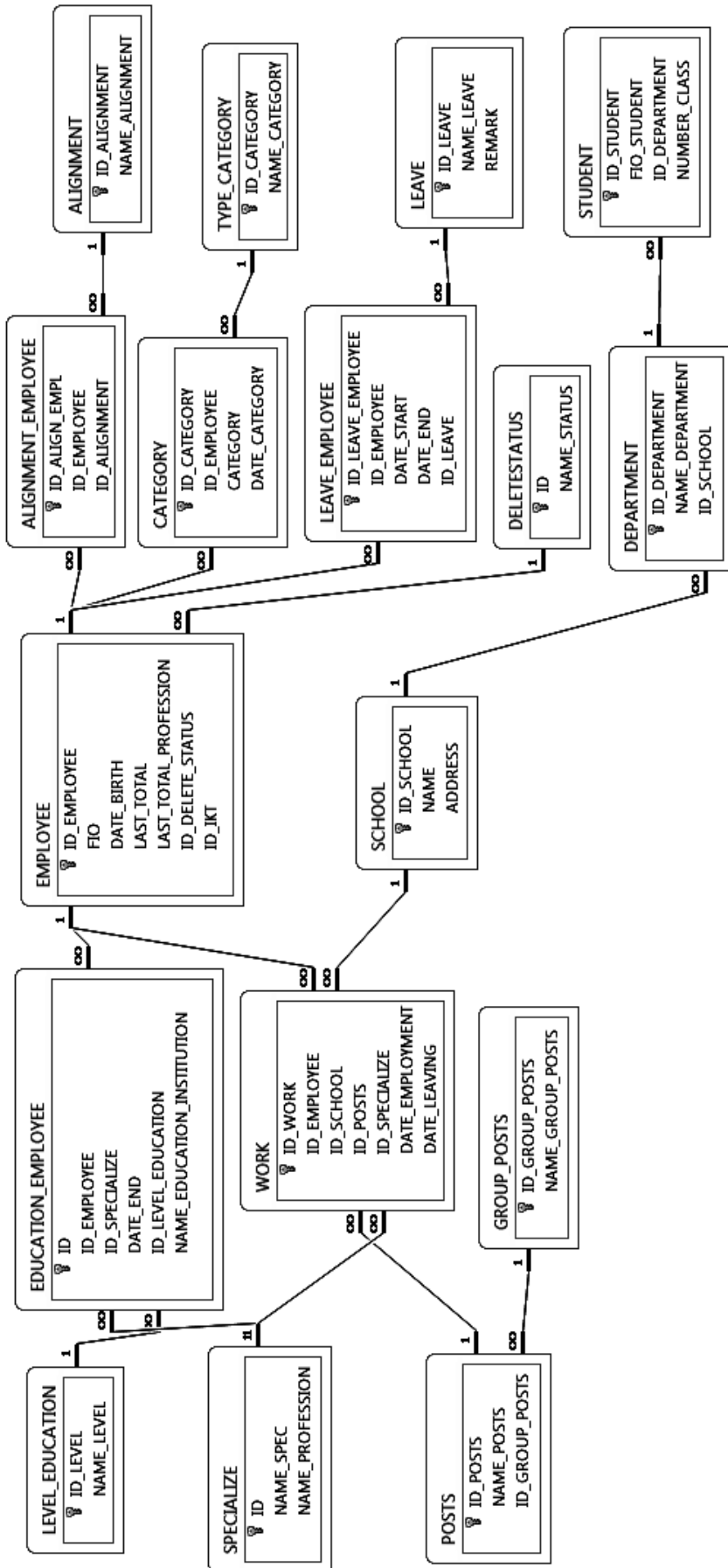


Рисунок 3.1 – Схема базы данных

Для дальнейшего использования базы данных и обращения к ней опишем используемые таблицы. В таблице 3.1 перечислены названия таблиц и описание их назначения.

Таблица 3.1 – Название и назначение таблиц базы данных

Название таблицы	Назначение
LEVEL_EDUCATION	Справочная таблица содержит информацию о видах образования (высшее, неоконченное высшее и т.д.)
SPECIALIZE	Справочная таблица содержит информацию о видах специализации, по которой сотрудник получил образование и работает в учреждении
POSTS	Справочная таблица содержит информацию о должностях.
GROUP_POSTS	Справочная таблица содержит информацию о группах должностей.
SCHOOL	Справочная таблица содержит информацию о филиалах школ.
DEPARTMENT	Справочная таблица содержит информацию об отделениях, которые прикреплены к филиалу школы.
TYPE_CATEGORY	Справочная таблица содержит информацию о видах категорий, которые может получить сотрудник
ALIGNMENT	Справочная таблица содержит информацию о видах совмещений сотрудников.
LEAVE	Справочная таблица содержит информацию о видах отпусков.

Окончание таблицы 3.1

Название таблицы	Назначение
DELETESTATUS	Справочная таблица содержит информацию о состоянии удаления записи.
EMPLOYEE	Содержит список сотрудников данного учреждения, а также их личную информацию
WORK	Содержит перечень должностей, занимаемых сотрудниками учреждения дополнительного образования.
EDUCATION_EMPLOYEE	Включает в себя перечень учебных заведений, в которых окончили обучение сотрудники, с указанием даты окончания и полученной специализации.
ALIGNMENT_EMPLOYEE	Содержит записи о совмещениях сотрудников
CATEGORY	Содержит записи о категориях, полученных сотрудниками
LEAVE_EMPLOYEE	Содержит список отпусков сотрудников.
STUDENT	Включает в себя список учащихся данного учреждения дополнительного образования.

В таблице 3.2 перечислены поля каждой таблицы, а также указаны их тип и назначение.

Таблица 3.2–Название и назначение полей таблиц базы данных

Название таблицы	Структура		Описание
	Поле	Тип данных	
Level_education	Id_level	integer	ID уровня образования
	Name_level	varchar(70)	Вид уровня образования
Specialize	ID	integer	ID специализации
	Nam_spec	varchar(20)	Название специализации
	Name_profession	varchar(100)	Название специальности
Posts	ID_Posts	integer	ID должности
	Nam_posts	varchar(20)	Название должности
	Id_group_posts	integer	ID группы должности
Group_posts	Id_group_posts	integer	ID группы должности
	Name_group_posts	varchar(70)	Название группы должности
School	ID_school	integer	ID школы
	Name	varchar(20)	Название школы
	Address	varchar(100)	Адрес школы
Department	ID_Department	integer	ID отделения
	Name_Department	varchar(20)	Название отделения
	ID_school	integer	ID школы
Type_category	Id_category	integer	ID категории
	Name_category	varchar(70)	Название категории

Продолжение таблицы 3.2

Название таблицы	Структура		Описание
	Поле	Тип данных	
Alignment	Id_alignment	integer	ID вида совмещения
	Name_alignment	varchar(70)	Название совмещения
Leave	ID_leave	integer	ID вида отпуска
	Name_leave	varchar(20)	Название вида отпуска
	Remark	varchar(200)	Примечание
Deletestatus	Id	integer	ID статуса удаления
	Name_status	varchar(15)	Название статуса удаления
Employee	Id_employee	integer	ID сотрудника
	FIO	varchar(70)	Фамилия, имя, отчество
	Date_birth	date	Дата рождения
	Last_total	varchar(10)	Предыдущий общий стаж
	Last_total_profession	varchar(10)	Предыдущий стаж по специальности
	Id_delete_status	integer	ID статуса удаления
	ID_IKT	integer	Имеет ли сотрудник подготовку по использованию ИКТ
Work	Id_work	integer	ID должности сотрудника
	ID_employee	integer	ID сотрудника
	ID_school	integer	ID школы

Продолжение таблицы 3.2

Название таблицы	Структура		Описание
	Поле	Тип данных	
	ID_posts	integer	ID_должности
	ID_specialize	integer	ID_специализации
	Date_employment	date	Дата трудоустройства
	Date_leaving	date	Дата увольнения с должности
Education_employee	Id	integer	ID должности сотрудника
	ID_employee	integer	ID сотрудника
	ID_specialize	integer	ID специализации
	Date_end	date	Дата окончания
	ID_level_education	date	ID уровня образования
	Name_education_institution	varchar(200)	Название учебного заведения
Alignment_employee	ID_align_empl	integer	ID совмещения сотрудника
	ID_employee	integer	ID сотрудника
	Id_alignment	integer	ID вида совместительства
Category	Id_category	integer	ID категории сотрудника
	ID_employee	integer	ID сотрудника
	Category	integer	ID категории
	Date_category	date	Дата присвоения категории

Окончание таблицы 3.2

Название таблицы	Структура		Описание
	Поле	Тип данных	
Leave_employee	Id_leave_employee	integer	ID отпуска сотрудника
	ID_employee	integer	ID сотрудника
	Date_start	date	Дата начала отпуска
	Date_end	date	Дата окончания отпуска
	ID_employee	integer	ID сотрудника
Student	Id_student	integer	ID ученика
	FIO_student	varchar(70)	Фамилия, имя, отчество ученика
	ID_department	date	ID_отделения
	Number_class	date	Номер класса

Из таблицы 3.2 видно, что в таблице с информацией о сотрудниках (таблица Employee) есть уникальный номер, который служит для идентификации записей о каждом сотруднике. Данный номер используется для связи с другими таблицами. Так как при расчете общего стажа и стажа по специализации необходимо учитывать стаж работы по предыдущим местам работы сотрудника, то в данной таблице предусмотрены поля для записи данных о предыдущем общем стаже (поле Last_total) и предыдущем стаже по специальности (поле Last_total_profession). Формат хранения в данных полях следующий:

«dd.mm.yy»,

Где dd – количество полных дней, засчитываемых в трудовой стаж;

mm – количество полных месяцев, засчитываемых в трудовой стаж;

yy – количество полных лет, засчитываемых в трудовой стаж.

Расчет стажа с учетом периодов работы в данном учреждении производится с помощью хранимой процедуры, алгоритм которой будет описан ниже.

Так как сотрудник может занимать не одну должность (сотрудник может выполнять работу на условиях внутреннего или внешнего совместительства), то данная таблица выделена отдельно (таблица Work). В данную таблицу заносится идентификатор сотрудника, филиал школы, в которую осуществляется прием сотрудника на работу, дата приема, дата увольнения. Также указывается должность, на которую принят сотрудник (ID_posts). Здесь следует уточнить, что каждая должность, имеющаяся в данном учреждении, относится к определенной группе: администрация, преподаватели, технический персонал. Данная информация занесена в справочную таблицу Group_posts.

При приеме на работу преподавателя указывается специализация преподаваемого им предмета (поле Specialize) . Кроме того, в таблице с информацией об образовании, полученном сотрудниками (таблица Education_Employee), также присутствует поле для указания полученной специальности (поле ID_specialize). Данные поля необходимы для расчета стажа сотрудника по специализации преподаваемого предмета.

Как было сказано выше, сотрудник может быть штатным работником и/или совместителем, поэтому для хранения данной информации предназначена таблица Alignment_employee, связанная с таблицей Employee с помощью идентификатора записи о сотруднике (ID_employee).

Также преподаватели периодически получают или подтверждают квалификационные категории. Информация обо всех категориях, полученных сотрудником, хранится в таблице Category, в которую также заносится дата присвоения категории.

Кроме того в базе данных учитываются отпуска, предоставляемые сотруднику, в том числе и по уходу за ребенком. Эти данные хранятся в отдельной таблице Leave_employee. Данная таблица также, как и таблицы Work, Alignment_employee, Category, Education_employee, связана с таблицей Employee с помощью уникального идентификатора сотрудника (ID_employee).

Также для обеспечения ссылочной целостности запись о сотруднике не удаляется, а в значение столбца `ID_delete_status` помещается значение статуса удаления, которое берется из справочной таблицы `DeleteStatus`.

Также в схеме базы данных расположены две таблицы: `Student` (данная таблица хранит информацию об учениках) и `Department` (таблица хранит информацию об отделениях, имеющих в данном учреждении дополнительное образования). Ученики в конце учебного года по результатам аттестаций либо переводятся в следующий класс, либо остаются в текущем. Поэтому необходимо отображать номер класса каждого ученика. Кроме того, возможно перемещение ученика во время обучения с одного отделения на другое, поэтому организована связь с таблицей `Department`.

Во время физического моделирования создается модель, оптимизированная для конкретного приложения и СУБД. На этом этапе рассматриваются основные отношения, организация файлов и индексов, предназначенных для обеспечения эффективного доступа к данным, а также все связанные с этим ограничения целостности и средства защиты.

На данной фазе проектирования выполнены следующие этапы:

- перенос логической модели данных в среду СУБД Firebird;
- реализация физической организации базы данных;
- разработка механизмов защиты базы данных;

Средствами защиты базы данных от несанкционированного доступа является разграничение прав доступа к объектам базы данных (таблицам, представлениям, хранимым процедурам, курсорам и т.д.). Данный метод осуществляется путем создания пользователей базы данных с паролем и определенной ролью, создание прав доступа и привилегий к каждому объекту базы данных для каждой роли. Данный способ был использован при создании базы данных.

Информация обо всех пользователях базы данных СУБД Firebird хранится в общей базе данных безопасности, которая располагается на сервере, и называется `security2.fdb`. Для работы с базой данных безопасности можно использо-

вать утилиту GSEC - это утилита для работы с базой данных безопасности. Она позволяет системному администратору управлять учетными записями пользователей для различных баз данных СУБД Firebird. Используя различные опции, можно добавлять, изменять или удалять учетные записи пользователей из базы данных безопасности.

Утилита GSEC, позволяет менять данные в таблице users базы данных безопасности. С ее помощью возможно добавление, изменение и удаление пользователей системы. Не все поля таблицы users могут быть показаны, хотя изменить можно любые из них. Например, столбец password (пароль) никогда не показывается утилитой GSEC, но его можно изменить.

3.3 Проектирование программного обеспечения

Разрабатываемая автоматизированная система должна обеспечить взаимодействие с данными, хранимыми в СУБД Firebird. На рисунке 3.2 изображена схема компонентов проектируемой программной системы.

Рассмотрим данную схему подробнее. На персональном компьютере установлена разрабатываемая автоматизированная система, которая обращается к ядру базы данных с помощью SQL-запросов. Данные, которые возвращает СУБД в результате выполнения запроса, передаются автоматизированной системе, обрабатываются и выводятся на монитор персонального компьютера.

Ядро базы данных является сердцевинной СУБД и выполняет следующие функции:

- отвечает за физическое структурирование и запись данных на диск;
- отвечает за физическое чтение данных с диска;
- принимает SQL-запросы от других компонентов СУБД (таких как генератор форм, генератор отчетов или модуль формирования интерактивных запросов), от пользовательских приложений и от других вычислительных систем.

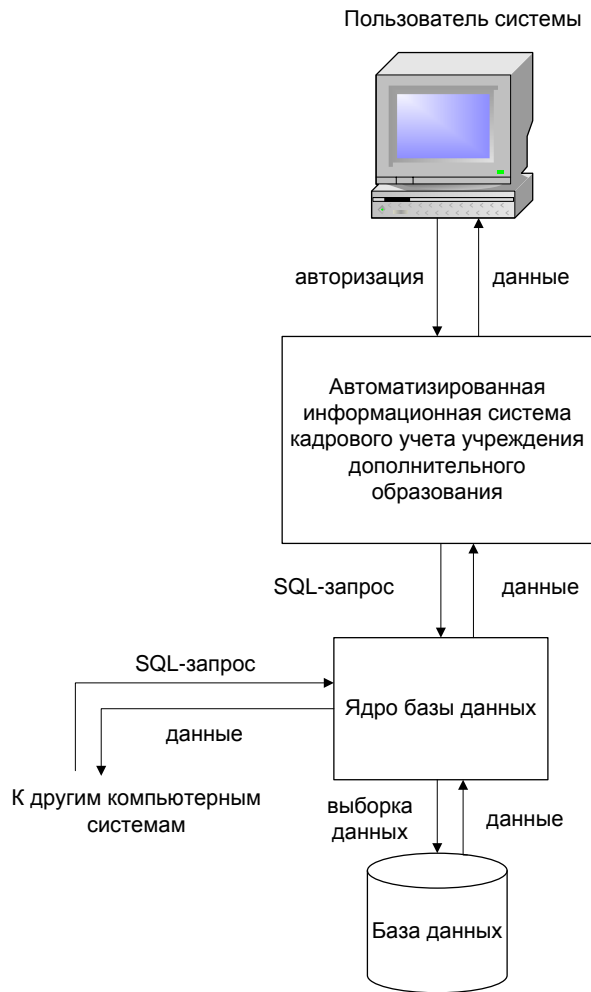


Рисунок 3.2 – Схема компонентов системы

Авторизация на сервере происходит по алгоритму, изображенному на рисунке 3.3.

При авторизации пользователь вводит логин и пароль и передает данные автоматизированной системе. Система осуществляет запрос подключения к базе данных. База данных проверяет: существует введенный логин и пароль в базе данных безопасности, и если существует, то определяет роль, к которой относится пользователь. Далее база данные возвращает результат выполнения запроса на авторизацию системе. Система, в случае успешного выполнения запроса, выполняет вход в систему с учетом роли пользователя. Если идентификацию пользователя не удалось выполнить, то пользователю выводится ошибка авторизации.

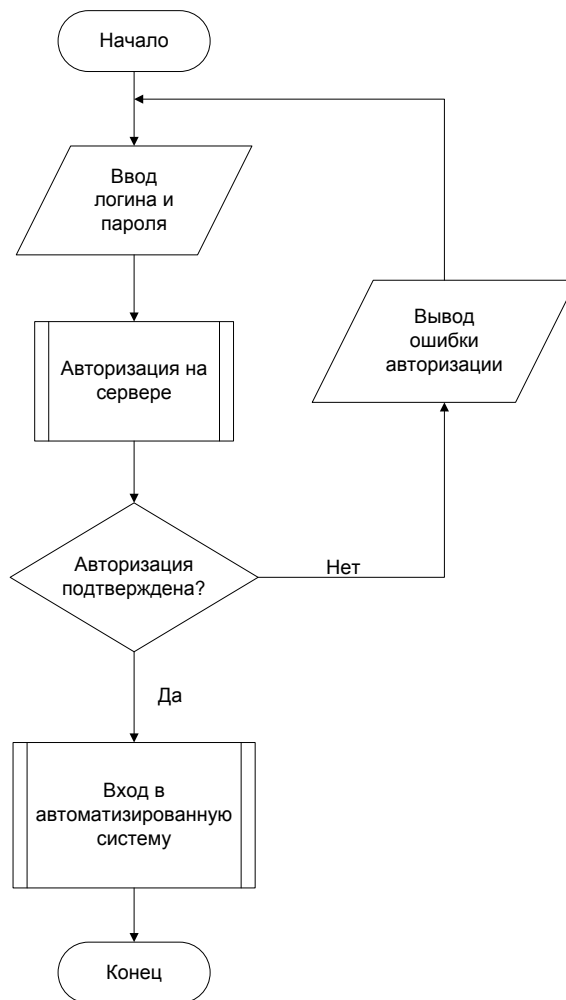


Рисунок 3.3 – Схема алгоритма авторизации на сервере базы данных

В СУБД Firebird можно создавать независимые модули, которые представляются разработчиками исходных кодов. Эти коды выполняются полностью на сервере, при необходимости возвращая приложению клиента значения, полученные в результате выполнения. Язык, который предоставляет такую возможность для сервера Firebird, называется PSQL (процедурный SQL). С помощью таких модулей появляется возможность распределения нагрузки на сервер базы данных при выполнении запросов от клиентских приложений.

Для вычисления стажа сотрудника разработана хранимая процедура на языке PSQL. Алгоритм вычисления стажа для временного диапазона приведен на рисунке 3.4.

При вычислении стажа необходимо учитывать, что количество дней в месяце принимается равным 30, и в году 12 месяцев. После вычисления стажа по перио-

дам работы необходимо просуммировать стаж в каждом периоде и прибавить полученную сумму к стажу сотрудника на предыдущем месте работы [10].

Работа с данными является одной из главных задач при создании как сетевых, так и автономных приложений. Библиотека .NET содержит богатый набор средств под общим названием ADO.NET(ActiveXDataObjects), поддерживающих взаимодействие с локальными и удаленными хранилищами данных.

Объектная модель ADO.NET состоит из классов двух видов: компоненты сущностей (content components) и компоненты управляемых поставщиков (managed-provider components). Основным классом первого вида является стандартный класс DataSet, представляющий собой набор связанных таблиц – локальную копию базы данных или ее части. Компоненты управляемых поставщиков обеспечивают интерфейс для доступа к данным (извлечения и обновления). Для непосредственной работы с данными используются стандартные классы Connection, Command и DataReader. Класс DataAdapter играет роль канала передачи данных между хранилищем и компонентами сущностей.

Для работы с ADO.NET необходимо подключить к проекту провайдер данных – стандартную динамическую библиотеку Firebird-Sql.Data.FirebirdClient.dll. Подключение к серверу базы данных осуществляется с помощью пространства имен Firebird-Sql.Data.FirebirdClient. Подключение к базе данных реализовано в разработанном классе ConnectionClass, в который передаются параметры подключения к базе данных. В данном классе также происходит заполнение необходимых визуальных компонентов данными, полученными в результате выполнения SQL-запроса. Описание используемых методов и полей данного класса описано ниже.

Программная система предусматривает экспорт данных в формат электронных таблиц Excel. Взаимодействие с Excel осуществляется с помощью COMInterop-технологии. Технология COM была задумана как механизм, при помощи которого программные компоненты получают информацию о возмож-

ностях других компонентов и обращаются к ним с запросами. Для доступа к объектам Excel используется библиотека `Microsoft.Office.Interop.Excel`.

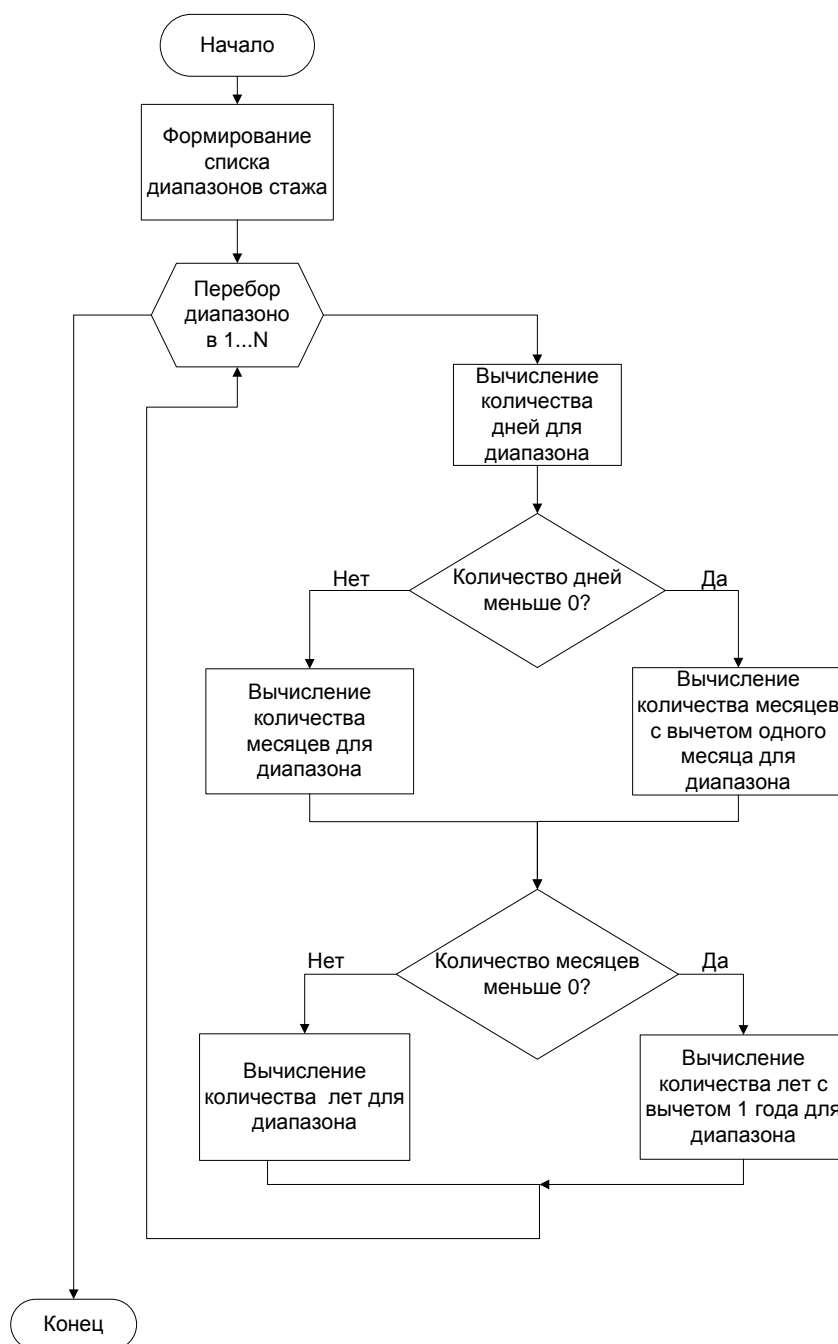


Рисунок 3.4 – Схема алгоритма вычисления стажа

3.4 Разработка программного обеспечения

Диалог пользователя с системой осуществляется с помощью графического интерфейса Windows – приложения. В основу Windows – приложений положен принцип событийного управления. То есть приложение после запуска ожидает действия пользователя и реагирует на них заранее заданным образом. Среда разработки VisualStudio .Net содержит удобные средства разработки Windows – приложения такие, как создание шаблонов приложения и форм, заготовок обработчиков событий, организацию циклов.

Для создания автоматизированной системы используется стандартное пространство имен `System.Windows.Form`, которое содержит большое количество стандартных классов, описывающих визуальные компоненты Windows – приложения.

Для сохранения и загрузки настроек подключения используется конфигурационный файл приложения (файл конфигурации) в VisualStudio .Net. Данный файл имеет формат XML, и при необходимости его можно изменить, если изменится настройка приложения.

Файл конфигурации состоит из элементов, которые являются логическими структурами данных, задающими сведения о конфигурации. Конфигурационный файл в разработанной автоматизированной системе состоит из двух секций:

- `configSections`, которая содержит все объявления обработчиков разделов настроек;
- `userSettings`, которая содержит пользовательские переменные для настройки программы, их тип и значение.

Реализация подключения к базе данных осуществлена в разработанном классе `ConnectionClass`, который содержит следующие поля:

- `private string user` – логин пользователя;
- `private string password` – пароль пользователя;
- `private string path` – путь к файлу базы данных;

– private FbConnectionStringBuilder fbCon – экземпляр класса, формирующего строку подключения;

– private FbConnection con – экземпляр класса, осуществляющего подключение к базе данных.

В конструкторе данного класса инициализируются поля user, password, path. При этом в переменную path сохраняется значение переменной из конфигурационного файла приложения.

Метод данного класса: private bool OpenConnection() осуществляет подключения к базе данных, в случае неудачи возвращает значение false.

Управление подключением к базе данных осуществляется с помощью стандартного класса FbConnection. Для формирования строки подключения используется класс FbConnectionStringBuilder.

Существует определенный набор свойств инициализации подключения к Firebird, который необходимо установить перед выполнением соединения с базой данных. Данные свойства и их описание указаны в таблице 3.3 [21].

Таблица 3.3 – Обязательные свойства инициализации подключения к Firebird.

Свойство	Описание
Location	Путь к базе данных на сервере.
Provider	Имя OleDb провайдера.
User ID	Имя пользователя базы данных.
Password	Пароль пользователя.
Stype	Кодировка определяет, символы какого национального алфавита будут использоваться [21].

Метод public bool VerificationConnection() – проверяет состояние подключения к серверу Firebird, в случае неудачи возвращает значение false.

Метод класса public bool CommandSelect(string query, DataGridView FdataGridView) – передает запрос, сохраненный в переменной query, базе данных и загружает результат выполнения запроса в таблицу FdataGridView. Листинг метода приведен ниже:

```

public bool CommandSelect(string query, DataGridView FDataGridView)
{
    if (OpenConnection())
    {
        FbCommand SelectSQL = new FbCommand(query, con);
        DataSet ds = new DataSet();
        //открываемтранзакцию;
        FbTransaction fbt = con.BeginTransaction();
        /*необходимопроинициализитьтранзакциюSelectSQL
        SelectSQL.Transaction = fbt;
        DataTable data = new DataTable();
        FDataGridView.DataSource = null;
        FbDataAdapter dataAdapter = new FbDataAdapter(SelectSQL);
        dataAdapter.Fill(data);
        FDataGridView.DataSource =
        data; /*передаемвисточникданныхтаблицу*/
FDataGridView.AllowUserToAddRows = false; /* пользователь не может
добавлять в таблицу строки*/
FDataGridView.ReadOnly = true; // таблица только для чтения
SelectSQL.Dispose(); //освобождаем ресурсы
con.Close();
//выравнивание ширины столбцов по содержимому всех ячеек
FDataGridView.AutoSizeColumnsMode =
DataGridViewAutoSizeColumnsMode.AllCells;
return true;
}
else
    return false;
}

```

Для того чтобы выполнить запрос к базе данных необходимо создать подключение к базе данных и открыть его. Это действие выполняет метод `OpenConnection()`. Далее создаем объект `FbCommand`, используя либо один из вариантов перегруженного конструктора, либо метод `FbConnection.CreateCommand()`. Затем создаем активную транзакцию из теку-

щего подключения – метод `con.BeginTransaction()` [21]. Для выполнения запроса необходимо проинициализировать транзакцию для объекта `FbCommand`.

Для хранения результата запроса используется экземпляр класса `DataTable`. Объект типа `DataTable` представляет собой таблицу базы данных. Объект `DataTable` состоит из строк и столбцов. Строки представляют собой отдельные записи таблицы, столбцы – соответствующие поля. После выполнения запроса экземпляр данного класса указывается в качестве источника данных для визуального компонента `DataGridViewFDataGrid`.

Объект `FbDataAdapter` – составная часть провайдера данных. То есть подсоединенная компонента объектной модели ADO .NET. Используется для заполнения объекта `DataTable` и модификации источника данных. Выполняет функции посредника при взаимодействии базы данных и объекта `DataTable`. После того как запрос выполнен, и данные занесены в таблицу `FDataGrid`, необходимо освободить все ресурсы, которые были задействованы при выполнении запроса, и закрыть подключение.

Метод `public bool CommandSelectComboBox(string query, ComboBox CB, List<string> id_table)` передает запрос, сохраненный в переменной `query`, базе данных и загружает результат выполнения запроса в выпадающий список `CB`, а идентификационные данные в переменную `id_table`. Данный метод используется для отображения данных из справочных таблиц.

Выполнение запроса происходит также как и в методе `CommandSelect(string query, DataGridViewFDataGrid)`, но вывод результата запроса выглядит следующим образом:

```
try
{
    DataReader = SelectSQL.ExecuteReader();
}
catch
{
    return false;
}
```

```

}
id_table.Clear();
try
{
    while (DataReader.Read())
    {
        id_table.Add(DataReader[0].ToString());
        CB.Items.Add(DataReader[1].ToString());
    }
}
finally
{
    DataReader.Close();
}

```

Метод `ExecuteReader()` возвращает объект `FbDataReader`. Навигация по строкам результирующего множества осуществляется при помощи метода `Read()`, который возвращает `true`, в случае если ещё остались строки и `false` в противном случае.

Метод `public bool CommandExecute(string query)` применяется для выполнения запросов, которые возвращают количество обработанных записей, таких как `insert`, `update`, `delete`, а так же для выполнения хранимых процедур, результат которых помещается в выходные параметры команды [21]. Выполнение запроса происходит аналогично методу `CommandSelect(string query, DataGridViewFDataGrid)`. Но при выполнении запросов, которые возвращают количество обработанных записей необходимо подтвердить транзакцию для её фиксации в базе данных:

```

int res = ExecuteSQL.ExecuteNonQuery(); /*для запросов, не
возвращающих набор данных необходимо вызывать этот метод*/
fbt.Commit(); /*если вставка прошла успешно подтверждаем
транзакцию*/

```

Также для экспорта в `MSEXcel` и печати содержимого справочных таблиц, информации о сотрудниках и сформированных статистических отчетов,

разработан собственный класс `ExportAndPrintTable`. В данном классе разработаны методы для вывода информации, хранимой в визуальном компоненте `DataGridView`, в электронную таблицу Excel. Поле данного класса является экземпляром визуального компонента `DataGridView` `fdgv`. Визуальный компонент `DataGridView` передается в класс `DataGridUser` с помощью свойства данного класса `DataGridValue`:

```
private DataGridView fdgv;
public DataGridView DataGridValue
{
    get
    {
        return fdgv;
    }
    set
    {
        fdgv=value;
    }
}
```

Объектная модель MS Excel имеет иерархическую структуру, в корне которой находится объект `Application` (`Excel.Application`), через который обеспечивается доступ к любой коллекции или внутреннему объекту приложения MS Excel или к компонентам открытых рабочих книг. Общая структура объектной модели MS Excel изображена на рисунке 3.5.

Основным элементом объекта «рабочая книга» является коллекция рабочих листов. Элементом коллекции является отдельный рабочий лист, который может представлять собой обычный лист (в виде таблицы) или быть в виде диаграммы. В последнем случае он ничего, кроме диаграммы, содержать не может.

Основной объект рабочего листа, с которым приходится работать, – ячейка. Ячейка как объект обладает множеством свойств и объектов, входящих в нее. Чтобы получить доступ к ним, необходимо получить доступ к самой ячейке, а затем изменять ее свойства. Ячейки объединены в области ячеек `Range`. Ячейки

объединены в строки и столбцы. Объединения строк и столбцов представляют собой коллекции, доступ к которым производится по числовому индексу или по буквенному обозначению столбца.

После завершения работы с объектом Excel необходимо уничтожить данный объект из памяти и вызвать сборщик мусора [22].

Так как формирование таблицы из кода занимает продолжительное время, то для данной цели используем шаблоны Excelc расширением .xlt.

Перечислим реализованные в данном классе методы и их назначение:

– `publicWorkbookExcelDataGridView(stringHeader)` – **ВЫВОДИТ** информацию из справочных таблиц в электронную таблицу Excel с указанием имени справочной таблицы и даты формирования документа;

– `public Workbook ExcelFormChisl()` **ВЫВОДИТ** отчет «Численность работников, состав преподавателей и концертмейстеров» в электронную таблицу Excel;

– `publicWorkbookExcelFormSost(stringdate)` **ВЫВОДИТ** отчет «Информация о кадровом обеспечении» в электронную таблицу Excel. Дата формирования отчета может задаваться пользователем;

– `privatevoidReleaseExcel(objectexcel)` **УНИЧТОЖАЕТ** объект excel из памяти и вызывает сборщик мусора;

– `public void PrintOtchet(string date,int numberOtch)` **ВЫВОДИТ** на печать сформированный отчет.

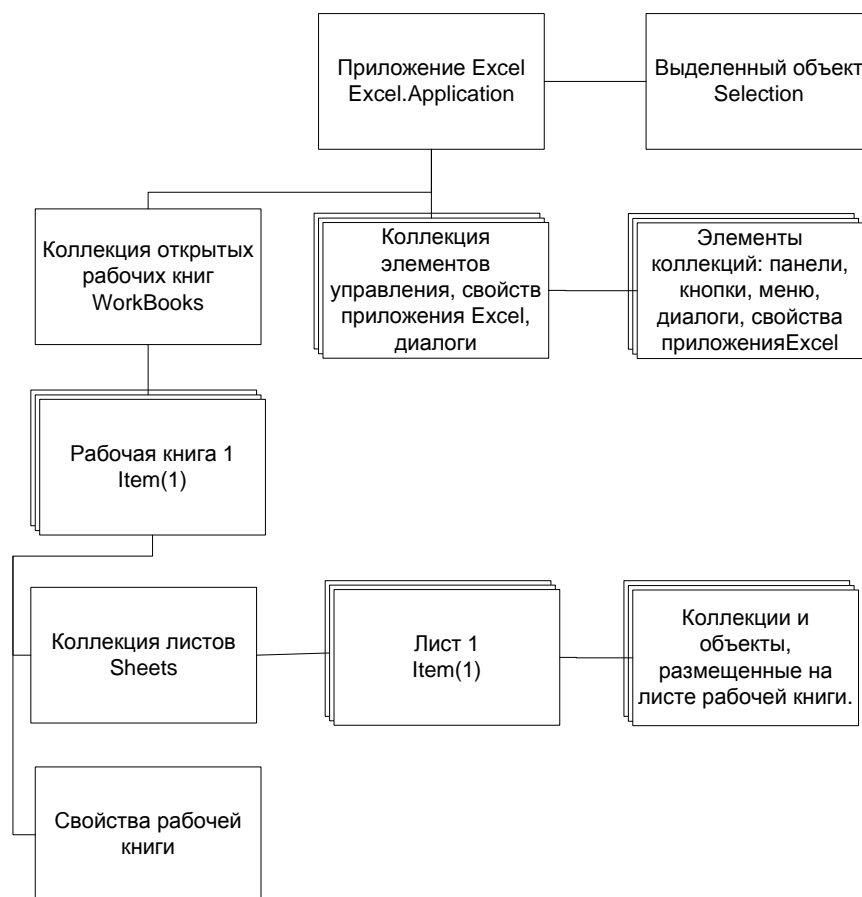


Рисунок 3.5 – Объектная модель MS Excel

Для отображения информации о сотрудниках необходимо сформировать SQL-запрос, который также будет выводить актуальную на текущий момент информацию об образовании, полученных категориях. Полученный SQL-запрос приведен ниже:

```

select em.id_employee, em.fio, em.date_birth,
(select * from FORMER_STAG(0,'12.05.2014',em.ID_Employee)),
(select * from FORMER_STAG(1,'12.05.2014',em.ID_Employee)),
decode(em.ID_IKT,0,'Да',1,'Нет','Неизвестно'),
em.id_delete_status, le_ed.name_level,
ed_em.name_education_institution, ed_em.date_end,
ty_cat.name_category, cat.date_category
from employee em left join education_employee ed_em
using (id_employee)
left join level_education le_ed
on (ed_em.id_level_education=le_ed.id_level)
left join category cat

```

```

using (id_employee)
left join type_category ty_cat
on (cat.category=ty_cat.id_category)
where ((ed_em.date_end =
    all (select max(ed_em2.date_end)
        from education_employee ed_em2
        where ed_em.id_employee=ed_em2.id_employee))
or (em.id_employee=
    any (select em3.id_employee
        from employee em3
            left join education_employee ed_emp3
            using (id_employee)
        where ed_emp3.id is null)))
and ((cat.date_category=
    all (select max(cat2.date_category)
        from category cat2
        where cat.id_employee=cat2.id_employee))
or (em.id_employee=
    any (select em3.id_employee
        from employee em3 left join category cat3
            using (id_employee)
        where cat3.id_employee is null)))

```

Так как информация о сотрудниках хранится в базе данных в разных таблицах, то для соединения таблиц используется «левое внешнее» соединение (leftjoin). При этом типе соединения в результат запроса будут включены все строки из левой таблицы и только те строки правой таблицы, для которых выполняется условие соединения. Для строк из левой таблицы, для которых не найдено соответствие из правой таблицы, в столбцы, извлекаемые из правой таблицы, заносится значение NULL.

В качестве условия вывода информации об образовании сотрудника является более поздняя дата окончания обучения в учебном заведении для выбранного сотрудника. Для извлечения строк из таблицы, содержащей информацию об образовании определенного сотрудника, используется внутренний запрос,

который производит выборку для всех записей из таблицы с информацией о сотрудниках. При этом во внутреннем запросе используется отличный от внешнего запроса псевдоним для таблицы employee. С помощью ключевого слова all производится сравнение результата запроса с максимальной датой окончания, причем результат сравнения должен быть истиной для всех значений, возвращаемых запросом.

Для выбора списка сотрудников, для которых отсутствует соответствующая запись в таблице с образованием также составлен внутренний запрос. С помощью предиката anupроизводится сравнение результата запроса и идентификатора сотрудника, причем результат сравнения должен быть истиной хотя бы для одного из списка значений.

На предыдущем листинге видно, что в запросе участвует вызов хранимой процедуры FORMER_STAG. Листинг процедуры приведен ниже:

```
create or alter procedure FORMER_STAG (  
    INDEXS integer,  
    DATEF date not null,  
    ID_EMPLOYEE integer not null)  
returns (  
    OUT_STRING varchar(35))  
as  
declare variable TEMPSTR varchar(10);  
begin  
    if (indexs=0) then  
        begin  
            for select * from stag(:datef,:id_employee)  
            into :tempstr  
            do  
                out_string='лет:'||substring(tempstr from 7)||  
                '; месяцев:'||substring(tempstr from 4 for 2)||  
                '; дней:'||substring(tempstr from 1 for 2);  
        end  
    if (indexs=1) then  
        begin
```

```

for select * from stag_spec(:datef,:id_employee)
into :tempstr
do
    out_string='лет:'||substring(tempstr from 7)||
    '; месяцев:'||substring(tempstr from 4 for 2)||
    '; дней:'||substring(tempstr from 1 for 2);
end
suspend;
end

```

Данная процедура служит для формирования строки общего стажа или стажа по специальности для определенного сотрудника, в зависимости от значения входной переменной `indexs`, и для даты заданной во входных параметрах процедуры. На данном листинге видно, что для формирования строки стажа происходит обращение к процедуре `stagi` или `stag_spec`. Данные процедуры непосредственно вычисляют общий стаж сотрудника или стаж по специальности. Общий алгоритм вычисления стажа для периодов работы в данном учреждении приведен на рисунке 3.4. Здесь приведем фрагмент кода процедуры `stag`.

```

ostmonth=0;
ostyear=0;
/* кол-во дней работы сотрудника в данном учреждении*/
days=days+daytemp;
if (days>29) then
    begin
        days=days-30;
        ostmonth=ostmonth+1;
    end
/* кол-во месяцев работы сотрудника в данном учреждении*/
months=months+monthtemp+ostmonth;
if (monthtemp>11) then
    begin
        months=months-12;
        ostyear=ostyear+1;
    end

```

```

end
/* кол-во лет работы сотрудника в данном учреждении*/
years=years+yeartemp+ostyear;
end
/* если есть стаж предыдущей работы, то учитываем его*/
if (laststag is not null) then
begin
/*общее количество проработанных дней*/
days=days+cast(substring(laststag from 1 for 2) as bigint);
if (days>29) then
begin
days=days-30;
ostmonth=ostmonth+1;
end
/*общее количество проработанных месяцев*/
months+=cast(substring(laststag from 4 for 2)as bigint)+ostmonth;
if (months>11) then
begin
months=months-12;
ostyear=ostyear+1;
end
/*общее количество проработанных лет*/
years=years+cast(substring(laststag from 7) as bigint)+ostyear;
if (days<10) then
temp_day_s='0' || cast(days as varchar(1));
else
temp_day_s=cast(days as varchar(2));
if (months<10) then
temp_month_s='0' || cast(months as varchar(1));
else
temp_month_s=cast(months as varchar(2));
formatstr= temp_day_s || '.' || temp_month_s || '.' || years;
end
suspend;
end

```

В данном фрагменте кода происходит суммирование всех промежутков работы сотрудника в данном учреждении. Как было сказано ранее, при расчете количество дней в месяце принимается равным 30. Если сумма превышает данную цифру, то в переменную `ostmonth` записывает дополнительный месяц и из полученного количества дней вычитается 30. Аналогично происходит расчет с месяцами, при учете, что в году 12 месяцев.

Также был реализован комбинированный поиск по сотрудникам, имеющимся в базе данных. На рисунке 3.6 показан интерфейс формы комбинированного поиска. На данном рисунке видно, что критерии поиска разбиты на 4 группы:

- фильтр по личным данным сотрудника;
- фильтр по полученному образованию;
- фильтр по занимаемым должностям;
- фильтр по трудовому стажу.

Исходя из схемы базы данных, изображенной на рисунке 2.1, данное разбиение является логичным.

В класс формы `CombineSearch` добавлены следующие поля:

`-privateString user, pass, path` - параметры подключения к базе данных (логин, пароль, путь);

`-publicString resultQuery` - строка запроса комбинированного поиска;

`-privateString select, from, where` - строки,

отвечающие за формирование секций `select, from, where` в результирующей строке `resultQuery`.

`-privateConnectionClass CC` - экземпляр класса подключения к базе данных.

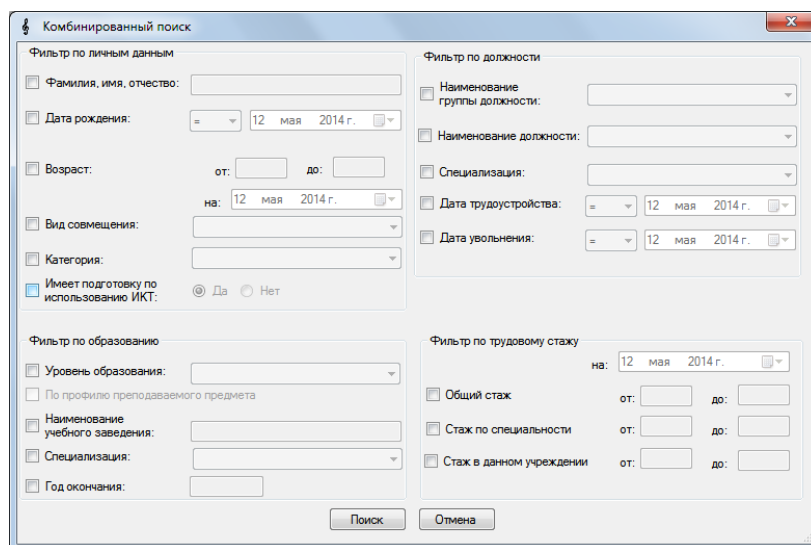


Рисунок 3.6 – Интерфейс формы комбинированного поиска

Загрузка списка справочных данных в визуальный компонент ComboBox происходит только при активации соответствующего элемента управления CheckBox (изменение свойства Enabled на значение true). Вызов метода заполнения comboBoxAlignment показан ниже:

```
if (comboBoxAlignment.Items.Count == 0)
    CC.CommandSelectCombobox("Select * from Aligment order by
    Name_Alignment", comboBoxAlignment, id_alignment);
```

Если в выпадающем списке нет элементов, то с помощью результата запроса из таблицы базы данных Aligment заполняется коллекция элементов comboBoxAlignment, а также в экземпляр класса List добавляются идентификаторы записей в базе данных для дальнейшего построения запроса.

Формирование запроса для поиска записей, удовлетворяющих критериям, заданным пользователями, осуществляет метод public void FormalizeStr(string FromAdd, string WhereAdd). Фрагмент кода данного метода представлен ниже:

```
if (FromAdd != null)
{
    List<string>ListRazd = newList<string>();
    string razd = "left join ";
    string Shablon = FromAdd;
```

```

bool flag = true;
while (flag)
{
    if (Shablon.Contains(razd))
    {
        int i = 0, j = 0;
        i = Shablon.IndexOf(razd);
        j = Shablon.IndexOf(razd, i + 10);
        if (j != -1)
        {
            ListRazd.Add(Shablon.Substring(i, j - i));
            Shablon = Shablon.Remove(i, j - i);
        }
        else
        {
            ListRazd.Add(Shablon.Substring(i))
            flag = false;
        }
    }
    else
        flag = false;
}

for (int inc = 0; inc < ListRazd.Count; inc++)
    if (from.IndexOf(ListRazd[inc].Trim())==-1)
        from += " " + ListRazd[inc];

}

if (WhereAdd != null)
    where += WhereAdd + " " + "and";

```

Так как в запросе на поиск предусмотрена возможность передачи нулевого параметра, то необходимо обработать этот случай. Далеевходная строка FromAdd, в которой хранятся именатаблицы, участвующих в запросе, разбивается на массив строк с помощью разделителя 'leftjoin'. Затем в цикле foreach проверяется сформированная секция запроса from на наличие в

ней добавляемых имен таблиц. В случае отсутствия имени оно добавляется к строке секции from. Соединение таблиц происходит с помощью «внешнего левого» соединения.

Формирование условий поиска (секция запроса where) происходит с помощью логической операции and, которая объединяет два логических выражения и возвращает значение true, если оба выражения имеют значение true.

Как видно на интерфейсе формы комбинированного поиска, в формировании запроса участвуют 4 вида поиска, то для реализации каждого вида поиска разработан свой метод:

```
- public void SelectedCheckBox(CheckBox Check, string FromS, string WhereS) – поиск по значению, заданному в текстовом поле;
```

```
- public void SelectedCheckWithCombo(CheckBox Check, ComboBox CB, List<string> id, string FromS, string WhereS) – поиск по значению, заданному в выпадающем списке;
```

```
- public void SelectedCheckWithDate(CheckBox Check, ComboBox CB, DateTimePicker DTP, string FromTable, string WhereS) – поиск по дате;
```

```
- public void SelectedCheckDiapozon(CheckBox Check, DateTimePicker DTP, TextBox TBWith, TextBox TBTo, string FromTable, string WhereStol, string wherestr) – поиск по временному диапазону.
```

Для поиска по временному диапазону используется встроенная функция `SQLDatediff(<временной_отрезок>, <значение1>, <значение2>)`, которая определяет величину временного промежутка от первого заданного значения типа дата, время или дата/время до второго. Возвращает значение типа `bigint`.

Также используется функция преобразования типов `cast(<выражение> as <тип данных>)`.

Листинг метода поиска по временному диапазону приведен ниже:

```
if (Check.Checked)
{
    string dateTime = DTP.Value.Day + "." + DTP.Value.Month +
        "."
```

```

+ DTP.Value.Year;
string diapazon = "datediff(year," + WhereStol +
",cast('" + dateTime + "' as date))";
if (TBWith.Text.Trim() != "" && TBTo.Text.Trim() != "")
{
    string whereDiap = "(" + diapazon + ">" + TBWith.Text
+ ") and (" + diapazon + "<" + TBTo.Text + ")";
    FormalizeStr(FromTable, wherestr + whereDiap);
}
else if (TBWith.Text.Trim() != "" && TBTo.Text.Trim() == "")

```

Вызов методов формирования строки поиска с параметрами приведен ниже:

```

SelectedCheckWithCombo(checkBoxSpec, comboBoxSpecEd, id_spec,
"education_employee ed_em", "(ed_em.ID_Employee=em.ID_employee)
and (ed_em.specialize");
SelectedCheckWithCombo(checkBoxNamePosts, comboBoxNamePosts,
id_posts, "work w", "(w.ID_Employee=em.ID_employee) and
(w.ID_Posts");
SelectedCheckWithCombo(checkBoxGroupPosts,
comboBoxNameGroupPosts, id_group, "work w, posts
p", "(w.ID_Employee=em.ID_employee) and (p.ID_posts=w.ID_Posts)
and (p.ID_group_posts");
SelectedCheckWithCombo(checkBoxSpecWork, comboBoxSpecWork,
id_spec_posts, "work w", "(w.ID_Employee=em.ID_employee) and
(w.ID_specialize");
SelectedCheckDiapozon(checkBoxAge, dateTimePickerAgeOn,
textBoxAgeWith, textBoxAgeTo, null, "em.Date_birth", "");

```

Продемонстрируем результат работы поиска. Для этого зададим параметры поиска, показанные на рисунке 3.7.

Рисунок 3.7 – Критерии поиска

В результате сформируется следующий запрос:

```

select em.id_employee,em.fio,em.date_birth,
       (select * from FORMER_STAG(0, '13.05.2014', em.ID_Employee)),
       (select * from FORMER_STAG(1, '13.05.2014', em.ID_Employee)),
       decode(em.ID_IKT,0, 'Да', 1, 'Нет', 'Неизвестно'),
       em.id_delete_status,
       le_ed.name_level,
       ed_em.name_education_institution,
       ed_em.date_end,
       ty_cat.name_category,
       cat.date_category
from (employee em left join education_employee ed_em
      using (id_employee))
left join level_education le_ed
on (ed_em.id_level_education=le_ed.id_level)
left join category cat
using (id_employee)
left join type_category ty_cat
on (cat.category=ty_cat.id_category)
left join work w
using (Id_Employee)
left join posts p
on (p.ID_posts=w.ID_posts)

```

Таблица с информацией о сотрудниках до выполнения операции поиска изображена на рисунке 3.8 (а). Результаты поиска показаны на рисунке 3.8 (б)

	Фамилия, имя, отчество	Дата рождения	Общий стаж	Стаж по специальности	Имеет подготовку по использованию ИКТ	Уровень образования
▶	Огородников Владимир Алексеевич	14.10.1970	лет:22; месяцев:10; дней:13	лет:22; месяцев:00; дней:17	Да	Высшее
	Ульянова Екатерина Владимировна	20.06.1808	лет:11; месяцев:00; дней:02	лет:11; месяцев:00; дней:02	Да	Высшее
	Третьяк Олег Евгеньевич	22.12.1989	лет:0; месяцев:10; дней:05	лет:0; месяцев:08; дней:13	Да	Высшее

Всего записей: 3

а)

	Фамилия, имя, отчество	Дата рождения	Общий стаж	Стаж по специальности	Имеет подготовку по использованию ИКТ	Уровень образования
	Ульянова Екатерина Владимировна	20.06.1808	лет:11; месяцев:00; дней:02	лет:11; месяцев:00; дней:02	Да	Высшее

Всего записей: 1

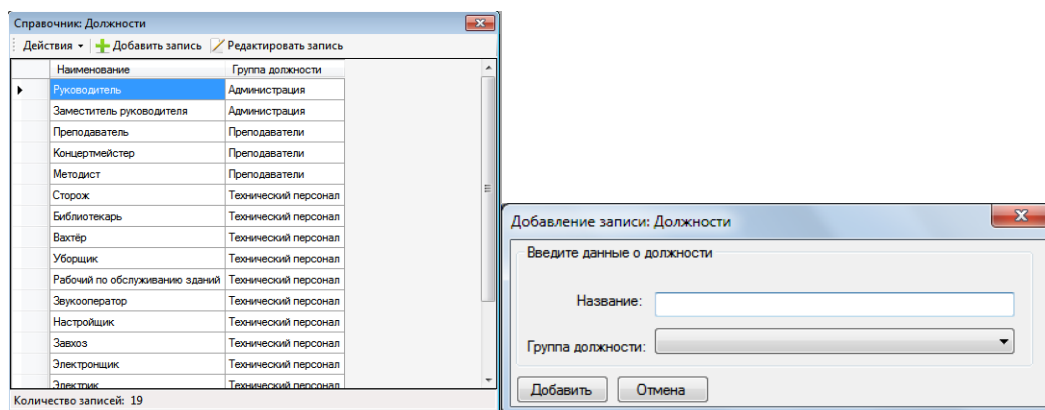
б)

Рисунок 3.8 – Результат выполнения комбинированного поиска

а) форма с информацией о сотрудниках до выполнения поиска;

б) форма с информацией о сотрудниках после выполнения поиска

Для корректного функционирования разрабатываемого программного обеспечения необходимо заполнить справочные таблицы спроектированной базы данных (справочники). Поэтому для них были разработаны интерфейсы. На рисунке 3.9 изображены формы отображения (а) и добавления записей (б) из справочника «Должности».



а)

б)

Рисунок 3.9–Интерфейс справочника «Должности»

а) форма отображения справочной информации;

б) форма добавления данных в справочную таблицу

В формах графического интерфейса справочников разработан метод `ViewDataGrid`, в который передается строка запроса. Метод устанавливает свойства для таблиц, в которые загружаются данные из справочников. Пример метода для формы отображения информации из таблицы `Posts` приведен ниже:

```
ConvertPath CP = new ConvertPath();
path = CP.FullPathConvert();
ConnectionClass CC = new ConnectionClass(user, pass);
CC.CommandSelect(query, dataGridView1);
dataGridView1.Columns[0].Visible = false;
dataGridView1.Columns[1].HeaderText = "Наименование".
```

Для добавления и редактирования предназначена одна и та же форма. При редактировании меняется название формы и визуального компонента кнопки. Также в форме присутствуют два конструктора:

– `public PostChangeForm(bool change, string Fuser, string Fpass)`. Данный конструктор вызывается для добавления данных. В него передаются параметры авторизации пользователя (логин и пароль);

– `public PostChangeForm(string FnamePosts, string Fuser, string Fpass, int Fid, int FidGroup)`. Данный конструктор вызывается для

изменения данных. В него передаются кроме параметров авторизации пользователя, также значения изменяемой строки и идентификационный номер записи.

Также в данной автоматизированной системе реализована возможность формирования отчетов по формам. Формирование отчета происходит на сервере базы данных в хранимых процедурах REPORTS_INF, которая принимает в качестве параметра номер отчета, который необходимо сформировать и дату на которую происходит формирование отчета. Разработанное программное обеспечение вызывает данную хранимую процедуру и после выполнения процедуры загружает полученные данные на форму. Листинг процедуры REPORTS_INF находится в приложении В. На рисунке 3.10 изображена форма отчета «Информация о кадровом обеспечении».

Состав специалистов	Всего	в т.ч. штат	в т.ч. совместители	Возраст до 30 лет (шт/совм.)	Возраст 30-55 лет (шт/совм.)	Возраст свыше 55 лет (шт/совм.)	Стаж до 5 лет (шт/совм.)	Стаж 6-10 лет (шт/совм.)	Стаж 11-25 лет (шт/совм.)	Стаж свыше 25 лет (шт/совм.)	Образование высшее (шт/совм.)	Образование среднее специальное (шт/совм.)
фортепиано	14	14	-	1/0	10/0	2/0	1/0	-	3/0	10/0	5/0	9/0
саксофон	2	2	-	-	2/0	-	-	2/0	-	-	-	2/0
труба	1	1	-	-	1/0	-	-	-	1/0	-	1/0	-
хоркласс	3	2	1	-	2/1	-	-	1/0	0/1	1/0	1/1	1/0
худкласс	1	-	1	-	0/1	-	-	-	0/1	-	0/1	-

Рисунок 3.10 – Отчет «Информация о кадровом обеспечении»

3.5. Описание модуля кадрового учета

Разработанный модуль кадрового учета предназначен для автоматизации кадрового учета МБОУ ДОД «ДМШ №2» ЗГО и автоматического формирования статистических отчетов. Программное обеспечение выполняет следующие функции:

- ввод, хранение, обработка информации справочных таблиц;
- ввод, хранение, обработка информации о сотрудниках учреждения дополнительного образования;
- поиск по заданным критериям информации о сотрудниках;
- формирование статистических отчетов о состоянии кадров в данном учреждении;
- вывод информации, содержащейся в базе данных в MS Excel или на печать;
- вывод сформированных статистических отчетов в MS Excel или на печать.

Минимальный состав используемых аппаратных средств персонального компьютера и их характеристики, необходимые для функционирования разработанного программного обеспечения:

- центральный процессор Intel Pentium 4 1.3GHz или выше;
- VGA совместимая видеокарта с 128 Мб памяти;
- монитор;
- накопитель на жестком магнитном диске: размер программного обеспечения может увеличиваться, в зависимости от размера файла базы данных. Минимум свободного дискового пространства: 30 Мб;
- объем ОЗУ не менее 256 Мб (рекомендуемый объем – 512 Мб);
- клавиатура;
- мышь.

Состав программных средств, необходимый для работы программного обеспечения:

- операционная система Windows XP или более поздней версии;
- сервер СУБД Firebird 2.0 или более поздней версии;
- .Net Framework 3.5 или более поздней версии;
- MS Office 2003 или более поздней версии.

Для печати отчетов требуется наличие принтера. Для работы с файлом базы данных, находящимся на другом компьютере, необходимо наличие настроенной локальной сети.

Для функционирования программного обеспечения, на каждый персональный компьютер, на котором планируется работа с разработанным программным продуктом, необходимо установить и настроить СУБД Firebird.

На компьютер, который будет выполнять роль сервера базы данных, необходимо скопировать файл базы данных, содержащий схему базы данных, описанную в пояснительной записке.

Установка программного продукта не требуется, достаточно скопировать папку с программой в любой каталог на жестком диске компьютера, где будет использоваться программа. После этого программный продукт полностью готов к работе.

Для выполнения программы необходимо запустить ее исполняемый файл. При загрузке программы запускается окно авторизации в системе, изображенное на рисунке 3.11.

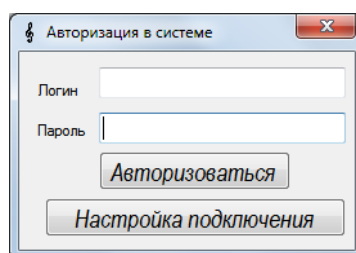


Рисунок 3.11 – Окно авторизации в системе

Данное окно содержит поля ввода логина и пароля пользователя и кнопку для запуска процесса авторизации на сервере базы данных. Также на форме рас-

положена кнопка для вызова окна настройки подключения к СУБД Firebird, представленное на рисунке 3.12.

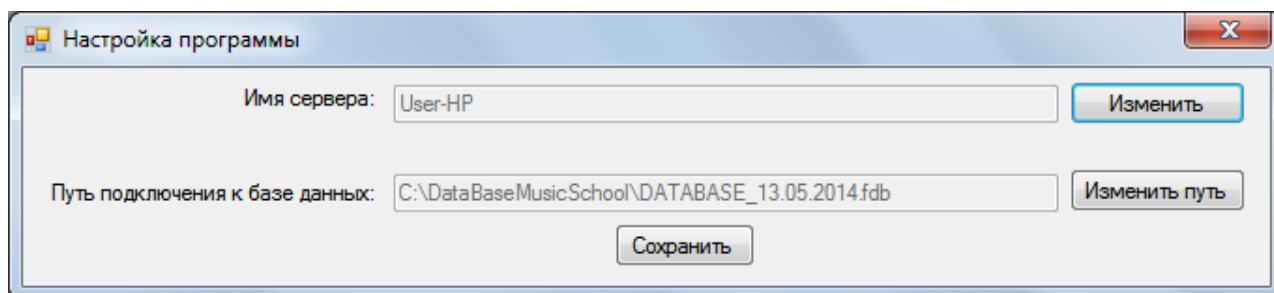


Рисунок 3.12 – Окно настройки подключения к базе данных

Настройка подключения к базе данных включает в себя два действия:

- указание имени сервера СУБД Firebird, на котором расположен файл базы данных;

- указание пути расположения файла базы данных на сервере СУБД Firebird.

При первоначальном запуске программы имя сервера и путь подключения к базе данных задан не будет.

В случае правильного указания местоположения файла базы данных, после нажатия на кнопку «Авторизоваться» (рис. 3.13), происходит процесс авторизации пользователя в системе, с указанными им логином и паролем.

При входе в систему с правами администратора, загружается главное окно, изображенное на рисунке 3.13.

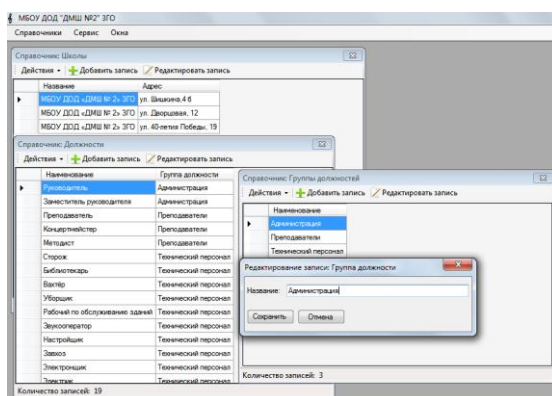


Рисунок 3.13 – Главное окно модуля администратора

При входе в систему с правами директора, загружается главное окно, изображенное на рисунке 3.14.

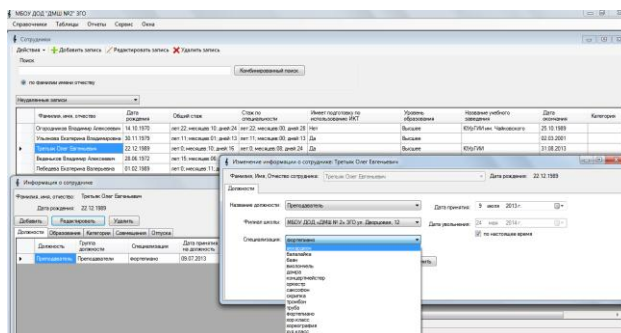


Рисунок 3.14 – Главное окно модуля директора

При входе в систему с правами бухгалтера, загружается главное окно, изображенное на рисунке 3.15.

№ строки	Наименование должностей	Всего работников на конец года, человек	Из гр.3 имеют подготовку по исполнительскому икт	Из гр.3 штатных работников	Из числа штатных работников, работающих на условиях штатного совместительства	Из гр. 5 имеют образование высшее профессиональное образование	Из них по профилю преподаваемого предмета	Из гр. 5 имеют образование среднее профессиональное образование	Из них по профилю преподаваемого предмета	Из гр. 5 имеют образование другое	Из гр. 5 имеют стаж до 5 лет	Из гр. 5 имеют стаж от 6 лет до 10 лет	Из гр. 5 имеют стаж от 11 лет до 25 лет	Из гр. 5 имеют стаж свыше 25 лет
1	2	3	4	5	6	7	8	9	10	11	12	13	14	15
41	Всего	4	3	4	-	3	3	1	1	-	1	-	3	-
	Из них:													
42	руководитель	1	1	X	X	1	X	-	X	X	-	-	1	-
43	заместитель руководителя	-	-	X	X	-	X	-	X	X	-	-	-	-
44	преподаватель	2	1	2	-	2	2	-	-	-	1	-	1	-
45	концертмейстер	1	1	1	-	-	-	1	1	-	-	-	1	-
46	методист	-	-	-	-	-	X	-	X	-	-	-	-	-

Рисунок 3.15 – Главное окно модуля бухгалтера

Вкладка меню «Справочники» главного окна– включает в себя список справочных таблиц. На рисунке 3.16 представлена данная вкладка.

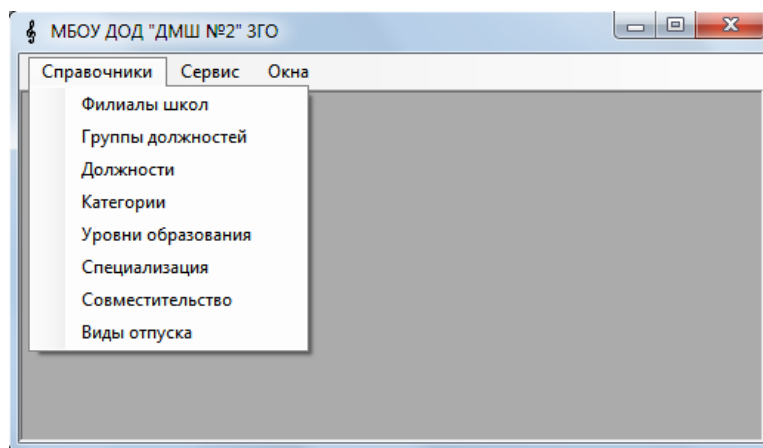


Рисунок 3.16 – Вкладка «Справочники»

При выборе пункта меню из категории вкладок «Справочники» загружается соответствующее окно, содержащее информацию из справочной таблицы. Работа со справочными таблицами будет продемонстрирована на примере таблицы «Должности». После выбора соответствующего пункта меню загрузится форма для отображения информации из справочной таблицы. Данное окно изображено на рисунке 3.17.

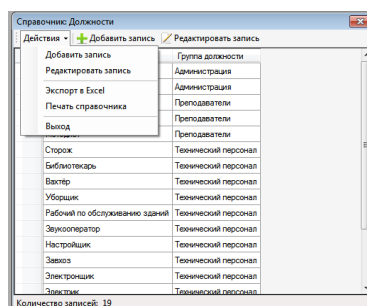


Рисунок 3.17 – Окно справочника «Должности»

Как видно на рисунке 3.17, для работы со справочниками предусмотрены следующие действия:

- добавить запись в справочную таблицу;
- редактировать выделенную запись;
- экспорт в Excel;
- печать справочника.

Рассмотрим выполнение данных действий подробнее.

Вызов окна добавления и редактирования записи из справочной таблицы можно произвести двумя способами:

- вызвав соответствующий пункт меню;
- нажав на соответствующую кнопку в панели инструментов.

Окно добавления информации в справочник изображено на рисунке 3.18. Для занесения информации в базу данных необходимо заполнить поля ввода, и нажать кнопку «Добавить». Для выхода из данного окна необходимо нажать кнопку «Отмена».

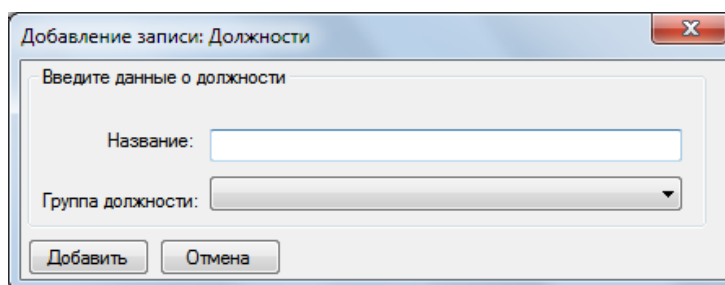


Рисунок 3.18 – Окно добавления записи в справочник

Для редактирования информации необходимо выделить соответствующую строку и выполнить действие для вызова окна редактирования записи. Окно для редактирования записи из справочника изображено на рисунке 3.19. Для фиксации изменения информации в базе данных необходимо нажать кнопку «Сохранить». Для выхода из данного окна необходимо нажать кнопку «Отмена».

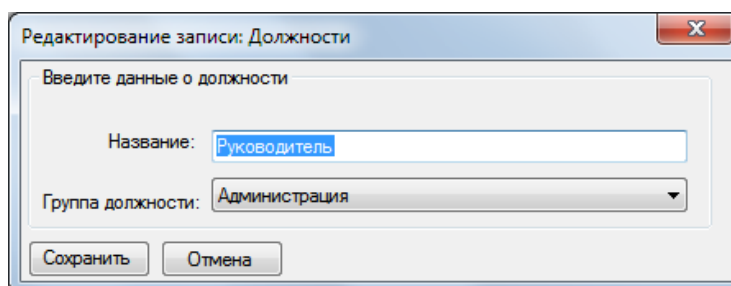


Рисунок 3.19 – Окно редактирования записи из справочника

Для экспорта данных в MS Excel и печати справочной таблицы необходимо выбрать соответствующий пункт меню. Окно экспорта в MS Excel, изображено

на рисунке 3.20, а предварительного просмотра перед печатью на рисунке 3.21. Работа с другими справочниками производится аналогично.

Экспорт данных | Режим совместности

Файл Главная Вставка Разметка страницы Формулы Данные Рецензирование Вид Раз

СайтВп 11 A A

Вставить Ж А Ч Ш

Буфер обмена Шрифт Выравнивание

	A	B	C	D	E	F
1	Справочник Должности					
2	Дата: 18.05.2014					
3	Наименование	Группа должности				
4	Руководитель	Администрация				
5	Заместитель руководителя	Администрация				
6	Преподаватель	Преподаватели				
7	Концертмейстер	Преподаватели				
8	Методист	Преподаватели				
9	Сторож	Технический персонал				
10	Библиотекарь	Технический персонал				
11	Вахтер	Технический персонал				
12	Уборщик	Технический персонал				
13	Рабочий по обслуживанию зданий	Технический персонал				
14	Звукооператор	Технический персонал				
15	Настройщик	Технический персонал				
16	Завхоз	Технический персонал				
17	Электронщик	Технический персонал				
18	Электрик	Технический персонал				
19	Заместитель директора по АХЧ	Технический персонал				
20	Секретарь	Технический персонал				
21	Бухгалтер	Технический персонал				
22	Главный бухгалтер	Технический персонал				
23						

Рисунок 3.20 – Окно MS Excel с экспортированными данными справочной таблицы

Предварительный просмотр | Экспорт данных | Режим совместности | - Microsoft Excel

Печать Параметры страницы Масштаб Показать поля Показать панель предварительного просмотра

Печать Масштаб Показать панель предварительного просмотра

Справочник Должности
Дата: 18.05.2014

Наименование	Группа должности
Руководитель	Администрация
Заместитель руководителя	Администрация
Преподаватель	Преподаватели
Концертмейстер	Преподаватели
Методист	Преподаватели
Сторож	Технический персонал
Библиотекарь	Технический персонал
Вахтер	Технический персонал
Уборщик	Технический персонал
Рабочий по обслуживанию зданий	Технический персонал
Звукооператор	Технический персонал
Настройщик	Технический персонал
Завхоз	Технический персонал
Электронщик	Технический персонал
Электрик	Технический персонал
Заместитель директора по АХЧ	Технический персонал
Секретарь	Технический персонал
Бухгалтер	Технический персонал
Главный бухгалтер	Технический персонал

Предварительный просмотр: страница 1 из 5 | Масштаб: 100%

Рисунок 3.21 – Окно предварительного просмотра

Для печати справочной таблицы необходимо выбрать действие «Печать» в окне предварительного просмотра.

Работа с другими справочными таблицами производится аналогично.

Вкладка «Сервис» главного окна программы – включает в себя список сервисных функций, которые может выполнить пользователь. На рисунке 3.22 представлена данная вкладка.

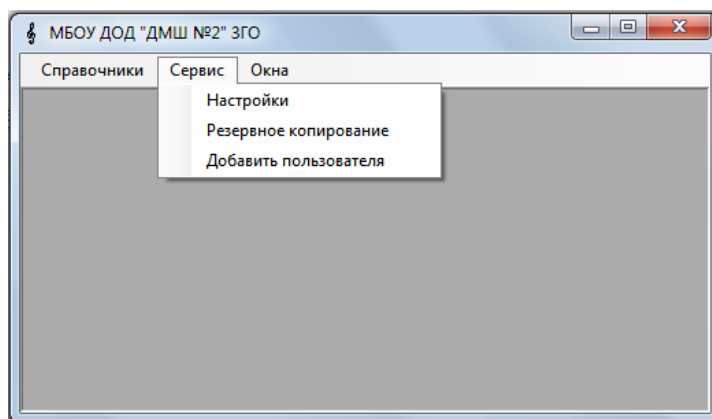


Рисунок 3.22 – Вкладка «Сервис»

При выборе пункта меню из данной категории вкладок загружается соответствующее окно, выполнение сервисных функций.

На рисунке 3.23, показана форма для резервного копирования данных.

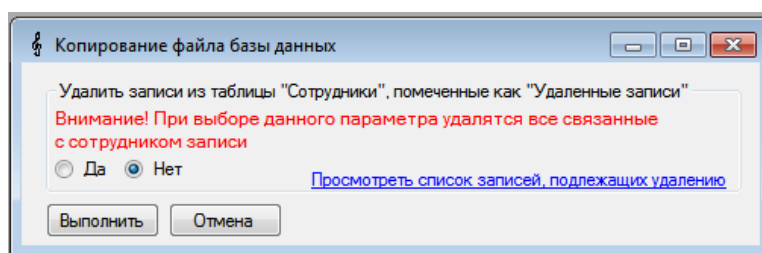


Рисунок 3.23 – Окно копирования файла базы данных

При выполнении функции резервного копирования существует возможность удаления всех записей о сотрудниках, помеченные как «Удаленные записи». Для просмотра записей, которые удалятся при выборе данного пункта, необходимо нажать на метку «Просмотреть список записей, подлежащих удалению». После нажатия на метку загружается форма, изображенная на рисунке 3.24. Для выхода из данного окна необходимо нажать кнопку «Отмена».

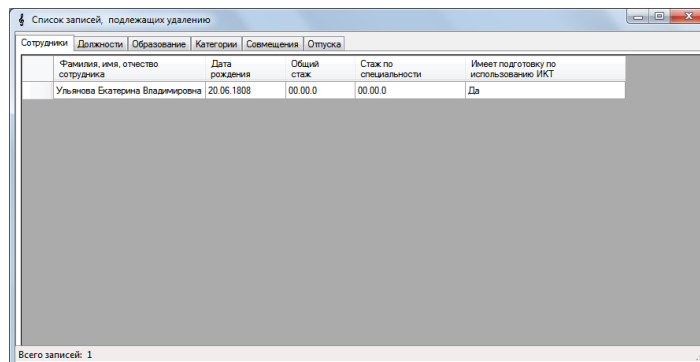


Рисунок 3.24 – Окно со списком записей, подлежащих удалению

Для выполнения резервного копирования необходимо нажать на кнопку «Выполнить». Копия файла базы данных будет располагаться в той же папке, где находится и основной файл базы данных. При этом к имени файла прибавится дата резервного копирования.

Окно для добавления нового пользователя базы данных, показано на рисунке 3.25.

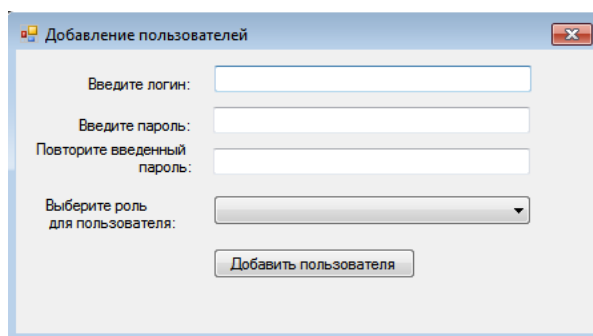


Рисунок 3.25 – Окно добавления пользователя

Для добавления пользователя необходимо заполнить все поля расположенные на форме и нажать на кнопку «Добавить пользователя».

Для вызова окна, которое отображает информацию о сотрудниках необходимо выбрать следующие пункты меню: «Таблицы» – «Сотрудники». После выполненных действий загрузится окно с информацией о сотрудниках, информация о которых занесена в базу данных. Данное окно показано на рисунке 3.26.

Фамилия, имя, отчество	Дата рождения	Общий стаж	Стаж по специальности	Имеет подготовку по использованию ИКТ	Уровень образования	Название учебного заведения	Дата окончания	Категория
Огородников Владимир Алексеевич	14.10.1970	лет:22, месяцев:10, дней:18	лет:22, месяцев:00, дней:22	Да	Высшее	Курский ин. Чайковского	25.10.1989	
Ульянова Екатерина Владимировна	20.06.1988	лет:11, месяцев:00, дней:07	лет:11, месяцев:00, дней:07	Да	Высшее		02.03.2001	
Трубник Ольга Евгеньевна	22.12.1989	лет:0, месяцев:10, дней:10	лет:0, месяцев:08, дней:18	Да	Высшее	Курский	31.08.2013	

Рисунок 3.26 – Окно с информацией о сотрудниках

Для работы с информацией о сотрудниках предусмотрены следующие действия:

- добавление записи в таблицу «Сотрудники»;
- просмотр подробной информации о сотруднике;
- добавление записи в таблицы, связанные с сотрудником, а именно: «Должности», «Образование», «Категории», «Совмещения»;
- редактирование информации о сотруднике;
- осуществление поиска по определенным критериям (функция комбинированного поиска);
- экспорт данных в Excel;
- печать данных из таблицы.

Рассмотрим выполнение данных действий подробнее.

Вызов окна добавления и редактирования записи о сотруднике можно произвести двумя способами:

- вызвав соответствующий пункт меню;
- нажав на соответствующую кнопку в панели инструментов.

Окно добавления информации о сотруднике изображено на рисунке 3.27. Для занесения информации в базу данных необходимо заполнить поля ввода, и нажать кнопку «Добавить». Для выхода из данного окна необходимо нажать кнопку «Отмена».

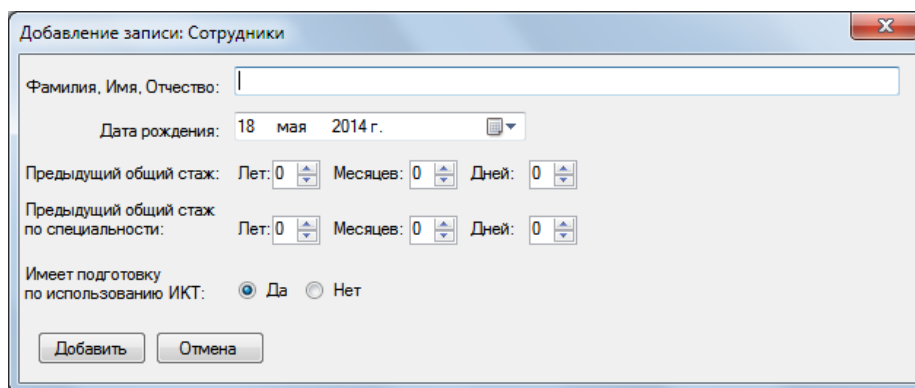


Рисунок 3.27 – Окно добавления информации о сотруднике

Для редактирования информации необходимо выделить соответствующую строку и выполнить действие для вызова окна редактирования записи. Окно для редактирования записи изображено на рисунке 3.28. Для фиксирования изменения информации в базе данных необходимо нажать кнопку «Сохранить». Для выхода из данного окна необходимо нажать кнопку «Отмена».

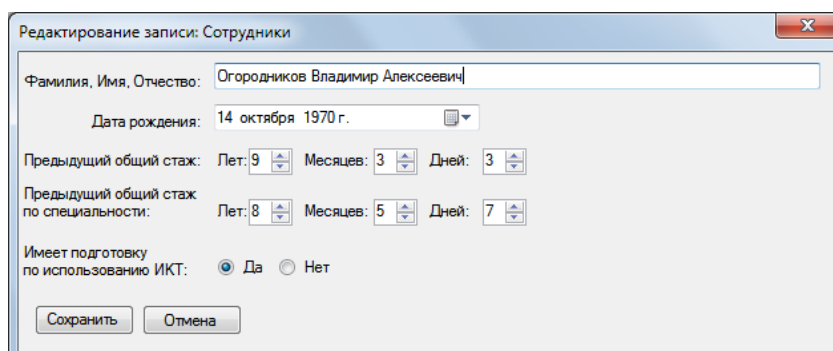


Рисунок 3.28 – Окно редактирования записи из справочника

Для просмотра подробной информации о конкретном сотруднике, необходимо выбрать запись в таблице с информацией о сотрудниках и щелкнуть на записи два раза манипулятором «мышь». Окно просмотра информации о сотруднике изображено на рисунке 3.29. Из данного окна также можно вызвать окно добавления и редактирования подробной информации о сотруднике. Для этого нужно выбрать необходимую вкладку: таблицу, в которую будет добавлена запись или из которой информация будет отредактирована. При нажатии на кнопку «Удалить» выбранная запись будет удалена из соответствующей таблицы.

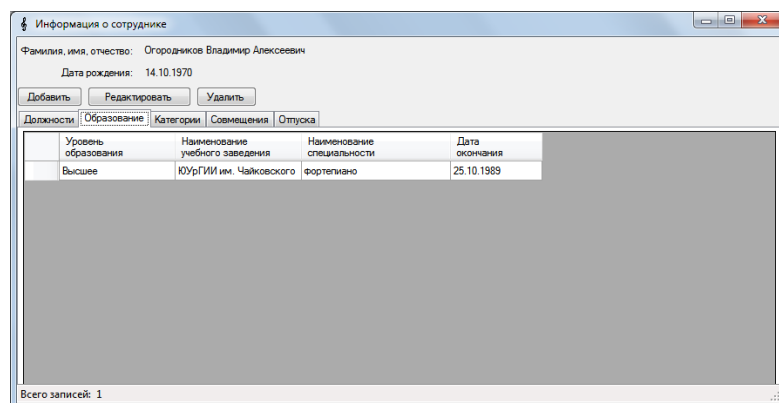


Рисунок 3.29 – Окно просмотра подробной информации о сотруднике

Также существует возможность комбинированного поиска по информации сотрудникам, имеющимся в базе данных. На рисунке 3.30 показан интерфейс формы комбинированного поиска. На данном рисунке видно, что критерии поиска разбиты на 4 группы:

- фильтр по личным данным сотрудника;
- фильтр по полученному образованию;
- фильтр по занимаемым должностям;
- фильтр по трудовому стажу.

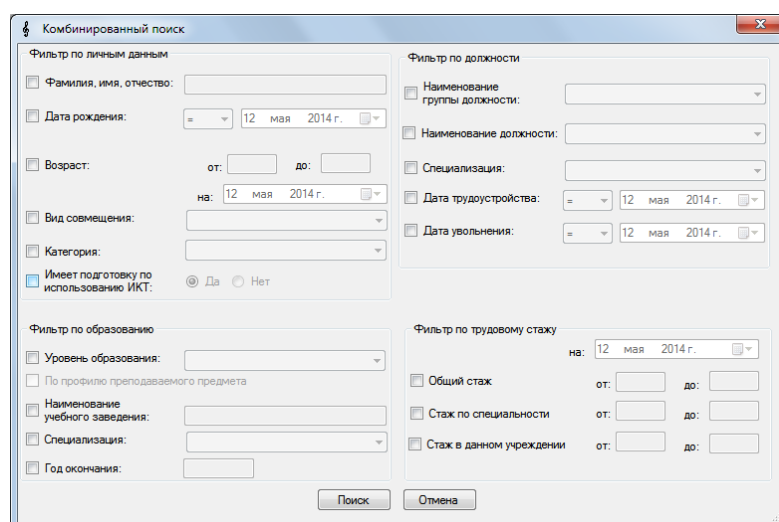


Рисунок 3.30– Интерфейс формы комбинированного поиска.

Для осуществления поиска по одному из параметров необходимо выбрать соответствующий флаг и задать критерий для поиска. После этого нажать на

кнопку «Поиск». После закрытия данного окна результат поиска отобразится в окне, показанном на рисунке 3.31.

Для экспорта данных в MS Excel и печати справочной таблицы необходимо выбрать соответствующий пункт меню. Данные функции выполняются также как и со справочными таблицами.

Вкладка «Отчеты» главного окна программы – включает в себя список отчетов, которые может сформировать пользователь. На рисунке 3.21 представлена данная вкладка.

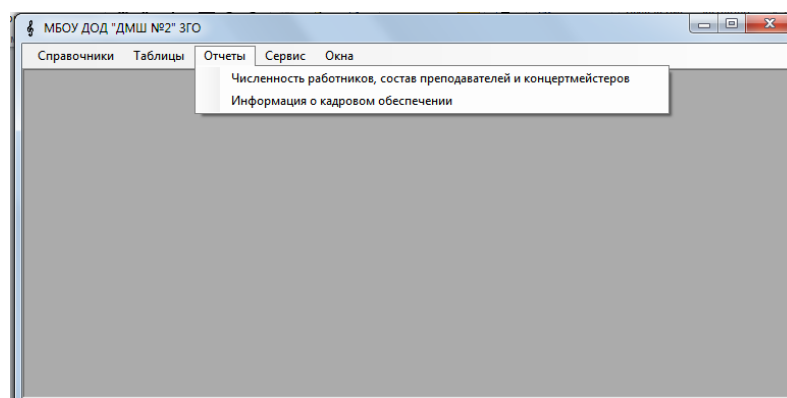


Рисунок 3.31 – Вкладка «Отчеты»

При выборе пункта меню из данной категории вкладок загружается соответствующее окно, содержащий отчет установленной формы. При загрузке отчета дата формирования равна текущей дате. На рисунке 3.32 изображены окна с отчетами «Численность работников, состав преподавателей и концертмейстеров» и «Информация о кадровом обеспечении». Как видно на рисунках, в данных формах отчета можно изменить дату формирования отчета, выбрав её из календаря. Также данные отчеты можно экспортировать в Excel и вывести на печать.

Численность работников, состав преподавателей и концертмейстеров														
Действия - Дата формирования отчета: 18 мая 2014 г.														
№ строки	Наименование должностей	Всего работников на конец года, человек	Из гр.3 имеют подготовку по использованию икт	Из гр.3 штатных работников	Из числа штатных работников, работающих на условиях штатного совместительства	Из гр. 5 имеют образование выше профессиональное образование	Из них по профилю преподаваемого предмета	Из гр. 5 имеют образование среднее профессиональное образование	Из них по профилю преподаваемого предмета	Из гр. 5 имеют образование другое	Из гр. 5 имеют стаж до 5 лет	Из гр. 5 имеют стаж от 6 лет до 10 лет	Из гр. 5 имеют стаж от 11 до 25 лет	Из гр. 5 имеют стаж свыше 25 лет
1	2	3	4	5	6	7	8	9	10	11	12	13	14	15
41	Всего	3	3	3	-	3	3	-	-	-	1	-	2	-
	Из них:													
42	руководитель	1	1	X	X	1	X	-	X	X	-	-	1	-
43	заместитель руководителя	-	-	X	X	-	X	-	X	X	-	-	-	-
44	преподаватель	2	2	2	-	2	2	-	-	-	1	-	1	-
45	концертмейстер	-	-	-	-	-	-	-	-	-	-	-	-	-
46	методист	-	-	-	-	-	X	-	X	-	-	-	-	-

Рисунок 3.32 – Окно «Численность работников, состав преподавателей и концертмейстеров»

Выводы по разделу три

В ходе разработки автоматизированной системы была спроектирована и реализована база данных, удовлетворяющая требованиям заказчика, разработан алгоритм работы программного обеспечения для управления спроектированной базы данных, выбраны средства реализации автоматизированной системы, реализовано программное обеспечение с выводом отчетов на печать и сохранением в электронном документе. Описан разработанный модуль кадрового учета для информационной системы детского учреждения дополнительного образования. Для разработанного модуля описано программное обеспечение, интерфейс.

ЗАКЛЮЧЕНИЕ

В настоящее время внедрение информационных систем в учреждения детского дополнительного позволит объединить все информационные потоки организации, исключить дублирования данных в различных структурных подразделениях, создаст эффективный инструмент для управления организацией.

Результатом выпускной квалификационной работы стало повышение эффективности управления кадрами в учреждении дополнительного образования. Для достижения этой цели была создан модуль кадрового автоматизированного учета кадров для МБОУДОД «ДМШ №2».

Для реализации системы были проанализированы документы учреждения дополнительного образования и существующие аналоги программ, автоматизирующих кадровый учет, выявлены их достоинства и недостатки. На основе полученной информации сформулированы требования к разрабатываемой автоматизированной системе. Решены вопросы хранения и обработки информации. При этом были решены следующие задачи:

- проведено теоретическое исследование предметной области, а также структуры и процесса организации детского дополнительного образования;
- выполнен анализ информационных систем, используемых в учреждениях дополнительного образования;
- спроектирован и разработан модуль кадрового учета Детской музыкальной школы №2.

Также была разработана программная документация: текст программного решения (приложение А).

БИБЛИОГРАФИЧЕСКИЙ СПИСОК

- 1 Абрамова, О.Ф., Аббазова, Р.А. Исследование проблем в области автоматизации управления узкоспециализированных малых предприятий / О.Ф. Абрамова, Р.А. Аббазова // Современная техника и технологии. – 2015. – № 1. – С. 43-48.
- 2 Афонин, А. М. Теоретические основы разработки и моделирования систем автоматизации: учебное пособие / А.М. Афонин, Ю.Н. Царегородцев, А.М. Петрова, Ю.Е. Ефремов. – М.: Форум, 2011. – 192 с.
- 3 Бакланов, А.В. Анализ состояния и проблемы информатизации управления образованием / А.В. Бакланов // Вестник кибернетики. – 2004. – №3. – С. 50-59.
- 4 Баронов, А.В. Информационные технологии и управление предприятием / В.В. Баронов, Г.Н. Калянов, Ю.И. Попов, И.Н. Титовский. — М.: Компания АйТи, 2006. — 328 с.
- 5 Варфоломеева, А.О. Информационные системы предприятия: учебное пособие / А.О. Варфоломеева, А.В. Коряковский, В.П. Романов. – М.: НИЦ ИНФРА-М, 2013. – 283 с.
- 6 Васильков, А.В. Информационные системы и их безопасность: учебное пособие / А.В. Васильков, А.А. Васильков, И.А. Васильков. –М.: Форум, 2013. – 528 с.
- 7 Володин, Д.В. Информационные технологии в современном документообороте: модель зрелости ЕСМ / Д.В. Володин. — Современные технологии делопроизводства и документооборота. 2012. — № 12. — С. 20-25.
- 8 Гагарина, Л.Г. Технология разработки программного обеспечения: учебное пособие / Л.Г. Гагарина, Е.В. Кокарева, Б.Д. Виснадул – М.: ИД «ФОРУМ»: ИНФА-М, 2008. – 400 с.
- 9 Голицына, О.Л. Основы проектирования баз данных. Учебное пособие / О.Л. Голицына, Т.Л. Партыка, И.И. Попов. – М.: Форум, 2014. – 416с.
- 10 Диго, С.М. Базы данных: проектирование и использование: учеб. для вузов / С.М. Диго. – М.: Финансы и статистика, 2005. – 592 с.

11 Заичкова, В.А. Автоматизированная информационно-аналитическая система «Управление образовательным учреждением» (АРМ Директор) - основа для формирования единого информационного пространства / А.В. Заичкова, И.Г. Лозницкий // Вестник Московского Городского Педагогического Университета. Серия. «Информатика и информатизация образования». – 2006. – №6. – С. 76-80.

12 Заботина, Н.Н. Проектирование информационных систем: учебное пособие / Н.Н. Заботина. – Братск: Изд-во БГУЭП, 2007. – Ч.1 –146 с.

13 Ислентьева, Е.В. Информационное обеспечение как средство повышения качества педагогического управления учреждениями дополнительного образования детей.: дис.... канд.пед.наук / Е.В. Ислентьева. – Екатеринбург, 2002 – 190 с.

14 Кодд, Е.Ф. Реляционная модель данных для больших совместно используемых банков данных. СУБД №1 /Е.Ф. Кодд. – М.: КноРус, 2005. –516 с.

15 Коровин, А.М. Новые информационные технологии в административном и корпоративном управлении: учебное пособие / А.М. Коровин. — Челябинск: Изд-во ЮУрГУ, 2003. — 113 с.

16 Кузнецов, С.Л. Новые Правила электронного документооборота — Дело-производство, 2015, — №1. — С.3–8.

17 Логиновский, О.В. Интеллектуальные информационные технологии и системы: учеб. пособие / О. В. Логиновский, В. М. Тарасов, Р. П. Чапцов; Челяб. гос. техн. ун-т. - Челябинск : Изд-во Челяб. гос. техн. ун-та, 1996. – 51 с.

18 Логиновский, О.В. Информационно-аналитическая система лица, принимающего стратегические решения/ О.В. Логиновский, В.В. Елагин // Информационные ресурсы России. - 2005. - № 3. - С. 12-16.

19 Логиновский, О.В. Повышение эффективности работы предприятия на основе современных управленческих технологий и информационно-аналитических систем / О.В. Логиновский, К.А. Коренная // Автоматизация и управление промышленными предприятиями: докл. науч.техн. конф. всерос. форума «Информационное общество-2015: вызовы и задачи». – Челябинск: Издат. центр ЮУрГУ, 2015. – С. 4–38.

20 Логиновский, О.В. Корпоративное управление: науч. изд. / О.В. Логиновский, А.А. Максимов. – М.: Машиностроение, 2007. – Т. 2. – 624 с.

21 Логиновский, О.В. Информационно-аналитические системы: концепции и методология создания / О.В. Логиновский, В.Н. Любицин // Информационно-аналитические компьютерные системы и технологии в региональном и муниципальном управлении: Научные труды. — Челябинск: Изд-во ЮУрГУ, ЦНТИ, РАЕН, 2001. — С. 41–52.

22 Логиновский, О.В. Построение систем электронного документооборота для органов управления: учебное пособие / О.В. Логиновский, В.В. Кокорюкин. — Челябинск: Изд-во ЮУрГУ, 2005. — 146 с.

23 Маркин, А.В. Построение запросов и программирование на SQL: учебное пособие/ А.В. Маркин – М.: Издательство «Диалог–МИФИ», 2008. – 20с.

24 О продукте «ГИСОФТ» – 2013. –http://gisoft.ru/?page_id=2.

25 Об утверждении статистического инструментария для организации Минкультуры России федерального статистического наблюдения за деятельностью учреждений культуры: Приказ Росстата от 15.07.2011 N 324 (ред. от 28.02.2013) // Консультант Плюс. Законодательство. Версия Проф / АО «Консультант Плюс». – М., 2013.

26 Об утверждении Правил подсчета и подтверждения страхового стажа для установления трудовых пенсий: Постановление Правительства РФ от 24.07.2002 № 555 (ред. от 21.04.2011) // Консультант Плюс. Законодательство. ВерсияПроф/ АО «Консультант Плюс». – М., 2013

27 Павловская, Т.А. С#. Программирование на языке высокого уровня: учебник / Т.А. Павловская. – СПб.: Питер, 2007. – 432 с.

28 Решения для функциональной задачи: Зарплата, управление персоналом и кадровый учет (HRM) / «1С» – 2014. –<http://solutions.1c.ru/hrm>.

29 Сотрудники организации «АраксГрупп»– 2013. –
<http://araxgroup.ru/content/view/35/65/>.

30 STUDENT INFORMATION SYSTEM «Custom – DB» – 2010. –
<http://www.custom-db.com/index.php/products/student-information-system>.

31 Сулейманов, А.Ю., Абрамова, О.Ф. Анализ проблем автоматизации бизнес-процессов многопрофильных образовательных учреждений / А.Ю. Сулейманов, О.Ф. Абрамова // Современная техника и технологии. – 2015. – № 6. – С. 53-58.

32 1С: Зарплата и кадры образовательной организации. – <https://solutions.1c.ru/catalog/edu-budghrm/features>.

33 Руководство по работе с БД Firebird с использованием библиотеки ADO .Net 2.0. – <http://citforum.ru/database/interbase/oledb/>.

34 Объектная модель Excel. – <https://susu.antiplagiat.ru/report/byLink/apiCorp/93819?v=1&userId=640&validationHash=B526546907953ABBF9DCA4E651530114857D8662&page=55>.

35 Язык программирования С# и платформа .Net Framework. – <http://www.avalon.ru/pps/C.NET/>.

36 Усманова И.В. Унификация документов — необходимая предпосылка внедрения СЭД / И.В. Усманова, Л.В. Коровина, Л.Р. Фионова // Делопроизводство. — 2014. — №1. — С.25-29.

37 Хейлсберг, А Язык программирования С#. 4-е изд. / А. Хейлсберг, М. Торгерсен, С. Вилтамут, П. Голд. – СПб.: Питер, 2012. – 784 с.

38 Шевченко, В.Н., Золотых, Н.Ю. Линейное и целочисленное линейное программирование / В. Н. Шевченко., Н. Ю. Золотых. – Нижний Новгород.: Издательство Нижегородского госуниверситета им. Н.И. Лобачевского, 2004. – 154 с.

39 Щеголев, А.Б. Разработка базы данных информационной системы учреждения дополнительного образования детей / А.Б. Щеголев, Ф.О. Федин, С.В. Чискидов, Е.Н. Павличева // Вестник РУДН. Серия «Информатизация образования». – 2015. – №1. – С. 101-118.

40 Янковая, В.Ф. Терминология в области управления документами / В.Ф. Янковская. — Современные технологии делопроизводства и документооборота. 2011. — С. 61-69.

ПРИЛОЖЕНИЕ А

Описание модуля кадрового учета

Разработанный программный продукт представляет собой автоматизированную информационную систему кадрового учета учреждения дополнительного образования.

Для хранения информации, необходимой для корректного функционирования системы, была выбрана СУБД Firebird. К преимуществам Firebird можно отнести многоверсионную архитектуру, обеспечивающую параллельную обработку оперативных и аналитических запросов, компактность (дистрибутив 5Mb), высокую эффективность и мощную языковую поддержку для хранимых процедур и триггеров. Для написания хранимых процедур использовался процедурный язык Firebird – PSQL.

В качестве языка программирования выбран объектно-ориентированный язык C# – это язык программирования, предназначенный для разработки самых разно-образных приложений, предназначенных для выполнения в среде .NET Framework. Он имеет поддержку компонентов, надежен и устойчив благодаря использованию "сборки мусора", обработки исключений, безопасности типов.

Для функционирования программы на компьютере должно быть установлено следующее программное обеспечение:

- операционная система Windows XP или более поздней версии;
- сервер СУБД Firebird 2.0 или более поздней версии;
- .Net Framework 3.5 или более поздней версии;
- MS Office 2003 или более поздней версии.

Функциональное назначение автоматизированной информационной системы:

- ввод, хранение, обработка информации справочных таблиц;
- ввод, хранение, обработка информации о сотрудниках учреждения дополнительного образования;
- поиск по заданным критериям информации о сотрудниках;

– формирование статистических отчетов о состоянии кадров в данном учреждении.

Условно программное обеспечение можно разбить на 4 модуля:

- хранимые процедуры, выполняемые в СУБД Firebird;
- подключение к СУБД Firebird и выполнение SQL-запросов из программного обеспечения, реализованного на языке С#;
- пользовательский интерфейс;
- экспорт и печать отчетов.

К хранимым процедурам относятся следующие процедуры:

- STAG. Данная процедура вычисляет общий стаж сотрудника, идентификатор которого является входным параметром для процедуры;
- STAG_SPEC. Процедура вычисляет стаж сотрудника по специальности, идентификатор сотрудника также, как и в предыдущей процедуре является входным параметром. Формат выходной строки в процедурах STAG и STAG_SPEC:

«dd.mm.yy»,

где dd – количество полных дней, засчитываемых в трудовой стаж;

mm – количество полных месяцев, засчитываемых в трудовой стаж;

yy – количество полных лет, засчитываемых в трудовой стаж;

- FORMER_STAG. Данная хранимая процедура преобразует формат стажа «dd.mm.yy», полученный из процедуры stag или stag_spec, в формат «лет:yy; месяцев: mm; дней: dd». Входными параметрами для данной процедуры являются идентификатор сотрудника, для которого вычисляется стаж, переменная indexs, определяющая вид стажа (общий стаж или стаж по специальности) и дата, на которую необходимо вычислить стаж;

- PARSE_STAG_YEAR. Процедура вычисляет полное количество лет стажа сотрудника;

– `PROCEDURE_AGE`. Данная процедура определяет количество сотрудников, имеющих возраст в определенных границах, которые задаются во входных параметрах. Также к входным параметрам относится идентификатор специализации, для которой необходимо выполнить данную процедуру;

– `PROCEDURE_STAG` Данная процедура определяет количество сотрудников, имеющих стаж в определенных границах, которые задаются во входных параметрах. Также к входным параметрам относится идентификатор специализации, для которой необходимо выполнить данную процедуру;

– `REPORTS_FORM_SPEC`. Процедура формирует данные для отчета о кадровом обеспечении для одной специализации, которая указывается во входном параметре процедуры;

– `REPORTS_INF`. Процедура формирует данные для отчета о кадровом обеспечении для всех специализаций, имеющих в справочной таблице специализаций;

– `RESULTFORMED`. Процедура формирует данные для отчета о численности работников, состава преподавателей и концертмейстеров для всех групп должностей, которые занесены в справочную таблицу;

– `RESULTFORMEDGROUPPOSTS`. Процедура формирует данные для отчета о численности работников, состава преподавателей и концертмейстеров для определенной группы должностей, которая передается в процедуру в качестве входного параметра.

Далее рассмотрим модуль подключения подключение к СУБД Firebird и выполнение SQL-запросов из программного обеспечения, реализованного на языке С#.

Подключение к серверу базы данных осуществляется с помощью пространства имен `FirebirdSql.Data.FirebirdClient`. Подключение к базе данных реализовано в разработанном классе `ConnectionClass`, в который передаются параметры подключения к базе данных.

В конструкторе данного класса инициализируются поля `user`, `password`, `path`. При этом в переменную `path` сохраняется значение переменной из конфигурационного файла приложения. Рассмотрим методы данного класса.

Метод данного класса: `private bool OpenConnection()` осуществляет подключения к базе данных, в случае неудачи возвращает значение `false`.

Управление подключением к базе данных осуществляется с помощью стандартного класса `FbConnection`. Для формирования строки подключения используется класс `FbConnectionStringBuilder`.

Метод `public bool VerificationConnection()` – проверяет состояние подключения к серверу `Firebird`, в случае неудачи возвращает значение `false`.

Метод `public bool CommandSelect(string query, DataGridView FDataGrid)` – передает запрос, сохраненный в переменной `query`, базе данных и загружает результат выполнения запроса в таблицу `FdataGrid`.

Метод `public bool CommandSelectCombobox(string query, ComboBox CB, List<string> id_table)` передает запрос, сохраненный в переменной `query`, базе данных и загружает результат выполнения запроса в выпадающий список `CB`, а идентификационные данные в переменную `id_table`. Данный метод используется для отображения данных из справочных таблиц.

Метод `public bool CommandExecute(string queru)` применяется для выполнения запросов, которые возвращают количество обработанных записей, таких как `insert`, `update`, `delete`, а так же для выполнения хранимых процедур, результат которых помещается в выходные параметры команды.

Пользовательский интерфейс реализован в виде графического интерфейса `Windows` – приложения. Пользовательский интерфейс представляет собой `Windows` – формы:

- добавления/редактирования информации справочных таблиц;
- отображения данных из справочных таблиц;
- добавление/редактирования личной информации сотрудников;

- отображение личной информации о сотрудниках;
- отчет «Численность работников, состав преподавателей и концертмейстеров»;
- отчет «Информация о кадровом обеспечении»;
- настройки подключения к базе данных;
- добавления пользователя системы;
- реализации функции резервного копирования с возможностью очищения базы данных от записей, помеченных на удаление;

Данные формы при определенных действиях пользователя (например, нажатие на кнопку или загрузка формы) взаимодействуют с классом `ConnectionClass`, выполняя определенные SQL-запросы.

Модуль экспорта и печати отчетов представлен классом `DataGridUser`. Для работы с редакторами MS Excel в приложение были подключены ссылки на сборки `Microsoft.Office.Interop.Excel`. Вывод документа на печать также реализован в MS Excel с функцией предварительного просмотра.

Минимальный состав используемых аппаратных средств персонального компьютера и их характеристики:

- центральный процессор Intel Pentium 4 1.3GHz или выше;
- VGA совместимая видеокарта с 128Мб памяти;
- накопитель на жестком магнитном диске: размер программного обеспечения может увеличиваться, в зависимости от размера файла базы данных. Минимум свободного дискового пространства: 30 Мб;
- объем ОЗУ не менее 256 Мб (рекомендуемый объем – 512 Мб).

К входным данным относятся информация об идентификационных данных пользователя (логин и пароль) для осуществления авторизации на сервере. Также к входной информации относятся данные о сотрудниках учреждения, справочная информация о видах филиалов школ, совместительства, категорий,

специализаций, специальностей, должностей, групп должностей, которые необходимо разместить в соответствующих таблицах базы данных.

К выходным данным относятся обработанная информация о сотрудниках, удовлетворяющая условиям выполнения SQL-запросов, формы отчетов «Численность работников, состав преподавателей и концертмейстеров» и «Информация о кадровом обеспечении».