

ФЕДЕРАЛЬНОЕ ГОСУДАРСТВЕННОЕ АВТОНОМНОЕ ОБРАЗОВАТЕЛЬНОЕ УЧРЕЖДЕНИЕ ВЫСШЕГО ОБРАЗОВАНИЯ
**«БЕЛГОРОДСКИЙ ГОСУДАРСТВЕННЫЙ НАЦИОНАЛЬНЫЙ
ИССЛЕДОВАТЕЛЬСКИЙ УНИВЕРСИТЕТ»
(НИУ «БелГУ»)**

ИНСТИТУТ ИНЖЕНЕРНЫХ ТЕХНОЛОГИЙ И ЕСТЕСТВЕННЫХ НАУК

Кафедра прикладной информатики и информационных технологий

**РАЗРАБОТКА АВТОМАТИЗИРОВАННОЙ СИСТЕМЫ «ОТДЕЛ КАДРОВ
ЦЕНТРА ГИГИЕНЫ И ЭПИДЕМИОЛОГИИ БЕЛГОРОДСКОЙ ОБЛАСТИ
НОВООСКОЛЬСКОГО РАЙОНА»**

Выпускная квалификационная работа бакалавра

**заочной формы отделения
направление подготовки 09.03.03 Прикладная информатика**

**4 курса группы 07001261
Кудрявцевой Анастасии Алексеевны**

Научный руководитель
Болгова Е.В.

БЕЛГОРОД 2016

СОДЕРЖАНИЕ

ВВЕДЕНИЕ	4
1 Аналитическая часть.....	7
1.1 Технико-экономическая характеристика предметной области.....	7
1.1.1 Характеристика предприятия.....	7
1.1.2 Краткая характеристика подразделения или видов его деятельности.....	9
1.2 Экономическая сущность задачи.....	13
1.3 Обоснование необходимости и цели использования вычислительной техники для решения задачи.....	14
1.4 Постановка задачи.....	15
1.4.1 Цель и назначение автоматизированного варианта решения задачи.....	15
1.4.2 Общая характеристика организации решения задачи на ЭВМ.....	16
1.5 Анализ существующих разработок и обоснование выбора технологии проектирования.....	18
2 Обоснование проектных решений.....	23
2.1 Обоснование проектных решений по техническому обеспечению.....	23
2.2 Обоснование проектных решений по информационному обеспечению.....	24
2.3 Обоснование проектных решений по программному обеспечению.....	26
2.4 Обоснование проектных решений по технологическому обеспечению.....	35
2.5 Обоснование выбора программных средств.....	37
3 Проектная часть.....	39
3.1 Информационное обеспечение задачи.....	39
3.1.1 Информационная модель и ее описание.....	39
3.1.2 Используемые классификаторы и системы кодирования.....	44

3.1.3 Характеристика первичных документов с нормативно-справочной и входной оперативной информацией.....	44
3.1.4 Характеристика базы данных.....	47
3.1.5 Характеристика результатной информации.....	48
3.2 Программное обеспечение задачи.....	49
3.2.1 Общие положения.....	49
3.2.2 Структурная схема пакета.....	51
3.2.3 Описание программных модулей.....	52
3.3 Технологическое обеспечение задачи.....	54
3.3.1 Организация технологии сбора, передачи, обработки и выдачи информации.....	54
3.3.2 Схема технологического процесса сбора, передачи, обработки и выдачи информации	56
3.4 Описание контрольного примера реализации проекта.....	58
4 Организационно-экономическая часть.....	71
4.1 Целесообразность разработки с экономической точки зрения.....	71
4.2 Калькуляция себестоимости научно-технической продукции.....	79
4.2.1 Расчёт трудозатрат на разработку программного продукта.....	79
4.2.2 Составление сметы затрат на разработку программного продукта.....	83
4.2.3 Обоснование цены программного продукта.....	86
4.2.4 Расчет экономической эффективности.....	86
ЗАКЛЮЧЕНИЕ.....	89
СПИСОК ИСПОЛЬЗОВАННЫХ ИСТОЧНИКОВ.....	93
ПРИЛОЖЕНИЯ.....	94

ВВЕДЕНИЕ

Бурное развитие информационных технологий и их внедрение во все сферы человеческой жизни в последние годы привели к резкому расширению понятия обрабатываемой в компьютерных системах информации. Сегодня немислимо представить себе деятельность предприятия без применения на нем информационных технологий. Качественно новый уровень развития экономики не может быть достигнут без эффективного использования персонала предприятий.

Кадровое планирование как инструмент целенаправленной и эффективной работы с персоналом является составной частью стратегии и развития предприятия при рыночных взаимоотношениях. По мере развития личности работника приходится все чаще согласовывать рыночные условия и интересы сотрудников предприятия. Развитие производства все в большей степени нуждается в планировании его кадрового обеспечения.

Разработка и совершенствование стимулов и мотивов к труду выходят за рамки научных и познавательных проблем и все больше ставятся в практическую плоскость как средство в борьбе с конкурентами. Специалисты по управлению "человеческими ресурсами" имеют комплексную систему знаний в области социологии, психологии и права, что позволяет им прекрасно разбираться в современных рыночных условиях, подбирать и формировать нужные кадры.

Актуальностью разработки автоматизированной информационной системы «Отдел кадров Центра гигиены и эпидемиологии Белгородской области Новооскольского района» является то, что деятельность отдела кадров связана с переработкой и хранением большого количества информации и документов. Большой документооборот на бумажных носителях замедляет работу, затормаживает эффективность доступа к нужной информации и затрудняет поиск информации. Из-за этого и возникает необходимость автоматизации процесса учета кадровой информации. Автоматизация позволит

сократить время необходимое для поиска нужной информации, ускорить документооборот.

Эффективность деятельности отдела кадров во многом зависит от созданного программного продукта. При разработке ПО специалист должен опираться на создание конкретной программы для решения поставленной задачи. Выполнение данных требований позволит снизить затраты на создание программного продукта и создать качественную систему позволяющую повысить экономическую эффективность, сократить трудоемкие расчеты.

Объектом исследования выпускной квалификационной работы является филиал ФБУЗ «Центр гигиены и эпидемиологии» Новооскольского района Белгородской области. Предмет исследования – процесс управления учетом информации о сотрудниках организации, их должностях и образовании, об оборудовании, которое числится за сотрудником.

Целью ВКР является совершенствование документооборота за счет автоматизации учета персональных данных о сотрудниках филиала «ФБУЗ Центр гигиены и эпидемиологии» Новооскольского района Белгородской области. Для достижения данной цели необходимо решить следующие задачи:

- выбор программного обеспечения для разработки АИС;
 - разработка информационных моделей с использованием программных и аппаратных средств;
 - разработка программного продукта;
 - тестирование и отладка разработанного программного продукта.
- Основные процессы жизненного цикла, входящие в состав проекта:
- обоснование требований к комплексу технических средств;
 - описание функциональной структуры;
 - анализ современных программных комплексов для решения поставленной задачи;
 - выбор инструментальных средств разработки;
 - описание архитектуры разрабатываемого продукта;
 - разработка внутренней структуры;

- программа и методика испытаний;
- создание эксплуатационной документации.

К предметной области проекта относятся числовые и текстовые типы данных. Источник данных, относящийся к данной разработке – штатное расписание филиала «ФБУЗ Центр гигиены и эпидемиологии» Новооскольского района Белгородской области. Основная функциональность предметной области – совершенствование документооборота за счет автоматизации учета персональных данных сотрудников учреждения. Организационная структура, имеющая отношение к проекту – отдел кадров.

Классическими примерами информационных систем кадрового менеджмента 1С:Предприятие 7.7, 1С:Предприятие 8.2 и другие.

В первой главе, посвященной анализу предметной области, будет рассмотрена характеристика объекта исследования филиала «ФБУЗ Центр гигиены и эпидемиологии» Новооскольского района Белгородской области, определена экономическая сущность задачи, обоснована необходимость использования вычислительной техники для решения поставленных в ВКР задач. Кроме того, в рамках первой главы будет проведен сравнительный анализ существующих разработок в области автоматизации кадрового учета.

Во второй главе будет проведено обоснование и выбор проектных решений по техническому, информационному и программному обеспечению разрабатываемой системы. А также проведено обоснование выбора программных средств.

В третьей главе будет разработано информационное обеспечение задачи, БД, программное обеспечение системы и ее технологическое обеспечение. Кроме того будет описан контрольный пример реализации проекта.

Четвертая глава представляет организационно-экономическую часть и будет посвящена расчету экономической эффективности разрабатываемой системы.

Выпускная квалификационная работа состоит из 126 страниц и содержит 29 рисунков, 6 таблиц и 7 приложений.

1 Аналитическая часть

1.1 Технико-экономическая характеристика предметной области

1.1.1 Характеристика предприятия

ФБУЗ «Центр гигиены и эпидемиологии» Белгородской области является некоммерческой организацией федеральным бюджетным учреждением здравоохранения Федеральной службы по надзору в сфере защиты прав потребителей и благополучия человека.

Центр входит в единую федеральную централизованную систему органов и учреждений, осуществляющих государственный санитарно-эпидемиологический надзор.

При осуществлении своей деятельности Центр руководствуется Конституцией Российской Федерации, федеральными конституционными законами, федеральными законами, актами Президента Российской Федерации и Правительства Российской Федерации, международными договорами Российской Федерации, актами Министерства здравоохранения и социального развития Российской Федерации, актами Федеральной службы, а также Уставом.

Задачами Центра является профессиональное и иное обеспечение деятельности Управления в пределах прав и полномочий, определенных Уставом.

Филиал ФБУЗ «Центр гигиены и эпидемиологии» Белгородской области Новооскольского района, имеющее в своем составе 7 филиалов, единственное учреждение в области, осуществляющее комплексную санитарно-эпидемиологическую оценку объектов и факторов среды обитания – это питьевая вода, почва, воздушная среда, вода открытых водоемов, пищевые продукты, продовольственное сырье, изделия промышленного и бытового назначения, строительные и полимерные материалы, биологические объекты.

Лабораторно-инструментальная база Центра позволяет проводить бактериологические, паразитологические, санитарно-химические, физико-химические, токсикологические, радиологические, вирусологические (в т.ч. на наличие генетически модифицированных источников) исследования, замеры физических факторов, заниматься дезинфекционной, дератизационной деятельностью, проводить санитарно-гигиеническое обучение, оказывать консультационные услуги по вопросам санитарно-эпидемиологического благополучия населения, защиты прав потребителей, соблюдения правил продажи отдельных видов товаров, выполнения работ, оказания услуг, по санитарно-эпидемиологической оценке условий труда.

Проведенные экспертизы, оценки, исследования обеспечивают уверенность в безопасности продукции. Лаборатории испытательного лабораторного центра учреждения прошли все этапы аккредитаций и внесены в единый реестр аккредитованных испытательных лабораторий Федеральной службы по аккредитации, дважды, в связи с вступлением в силу Технических Регламентов Таможенного Союза лаборатории центра проходили дополнительную аккредитацию – в феврале на проведение исследований промышленной продукции (группа ТР ТС – изделия легкой промышленности, парфюмерно-косметической продукции, детских изделий, спецодежды и т.д.) и в сентябре 2013 на проведение исследований по группе по пищевой продукции в объеме ТР ТС. Кроме того, с ноября 2013 года лаборатории центра внесены в реестр аккредитованных организаций, оказывающих услуги в области охраны труда (т.е. проведение аттестации рабочих мест, а с принятием нового ФЗ специальной оценки условий труда), также центр имеет лицензии на деятельность с микроорганизмами 2-4 групп патогенности и на медицинскую деятельность. Результаты испытаний, исследований принимаются на всей территории Российской Федерации и Таможенного союза.

1.1.2 Краткая характеристика подразделения или видов его деятельности

Слово «учет» подразумевает прием сотрудника на работу, отслеживание его состояния в течение периода его работы и, наконец, увольнение сотрудника с работы. Дополнительно, отдел кадров должен формировать много отчетов перед бухгалтерией и внешними организациями, например, Пенсионным фондом и статистическими органами. Работа отдела кадров регламентирована законодательством и коллективным договором. В частности, predeterminedены формы приказов о приеме и увольнении, личные карточки сотрудников, формы отчетов.

Функции отдела кадров включают в себя:

- 1) Прием на работу новых сотрудников.
- 2) Увольнение сотрудников.
- 3) Кадровое перемещение:
 - изменение категории;
 - изменение должности/специальности;
 - изменение разряда;
 - изменение графика работы;
 - перевод на другой объект.
- 4) Воинский учет.
- 5) Учет членства в профсоюзе.
- 6) Оформление отпусков.
- 7) Оформление больничных листов.
- 8) Ведение штатного расписания.
- 9) Отчетность:
 - перед бухгалтерией;
 - перед Пенсионным фондом;
 - перед статистическими органами;
 - перед руководством.

Процесс моделирования информационной системы в IDEF0 начинается с определения контекста, то есть наиболее абстрактного уровня описания системы в целом. В контекст входят определение субъекта моделирования, цели и точки зрения на модель.

В качестве субъекта выступает деятельность отдела кадровфилиала ФБУЗ«Центр гигиены и эпидемиологии» Новооскольского района Белгородской области по учету сотрудников предприятия. Цель моделирования – анализ бизнес-процессов деятельности отдела кадровфилиалаФБУЗ «Центр гигиены и эпидемиологии» Новооскольского района Белгородской области по учету сотрудников предприятия. В качестве точки зрения модели выбрана точка зрения «Сотрудник отдела кадров филиала «ФБУЗЦентр гигиены и эпидемиологии» Новооскольского района Белгородской области».

При описании бизнес-процессов выбран уровень детализации, при котором по экспертной оценке работников ОК подробно описываются потоки основные потоки информации, связанные с деятельностью инспектора ОК.

При моделировании согласно нотации IDEF0 используются следующие понятия:

- Работы. Обозначают именованные процессы, функции или задачи, которые происходят в течение определенного времени и имеют распознаваемые результаты. Работы изображаются в виде прямоугольников. Все работы должны быть названы и определены.

- Взаимодействие работ с внешним миром и между собой описывается в виде стрелок. Стрелки представляют собой некую информацию и именуется существительными. В IDEF0 различают пять типов стрелок:

- вход (input) - материал или информация, которые используются или преобразуется работой для получения результата (выхода);

- управление (Control) - правила, стратегии, процедуры или стандарты, которыми руководствуется работа;

- выход (Output) - материал или информация, которые производятся работой;

- механизм (Mechanism) - ресурсы, которые выполняют работу;
- вызов (Call) - специальная стрелка, указывающая на другую модель работы.

На рисунке 1.1 приведена контекстная диаграмма для модели бизнес-процессов деятельности отдела кадров филиала «ФБУЗ Центр гигиены и эпидемиологии» Новооскольского района Белгородской области по учету сотрудников.

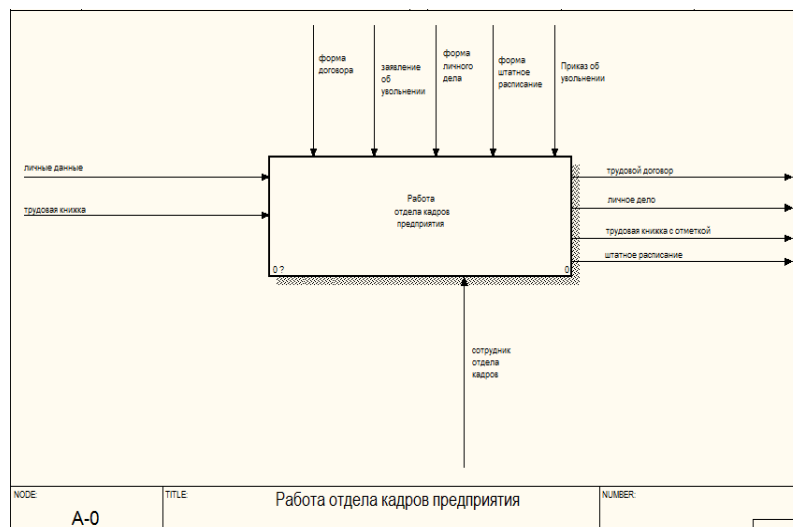


Рисунок 1.1 – Контекстная диаграмма деятельности отдела кадров филиала «ФБУЗ Центр гигиены и эпидемиологии» Новооскольского района Белгородской области по учету сотрудников

Контекстная диаграмма представляет собой самое общее описание системы и ее взаимодействия с внешней средой.

На вход основной функции, изображенной на контекстной диаграмме, подаются:

- личные данные сотрудников;
- трудовая книжка;

В качестве управления на систему в целом действуют:

- форма договора;
- заявление об увольнении;
- форма личного дела;

- форма штатного расписания;
- приказ об увольнении

Механизмами являются:

– инспектор отдела кадров, который выполняет основную работу по сбору и анализу информации, подготовке и оформлению документов, формированию отчетной документации.

После преобразования входной информации, при воздействии управления с помощью описанных выше механизмов получаем выходную информацию, которой для системы являются различные формы отчетов по запросам руководства:

- трудовой договор;
- личное дело;
- трудовая книжка с отметкой;
- штатное расписание;

Далее выполняется разбиение (детализация) контекстной диаграммы на диаграммы декомпозиции.

На рисунке 1.2 приведена диаграмма декомпозиции контекстной диаграммы.

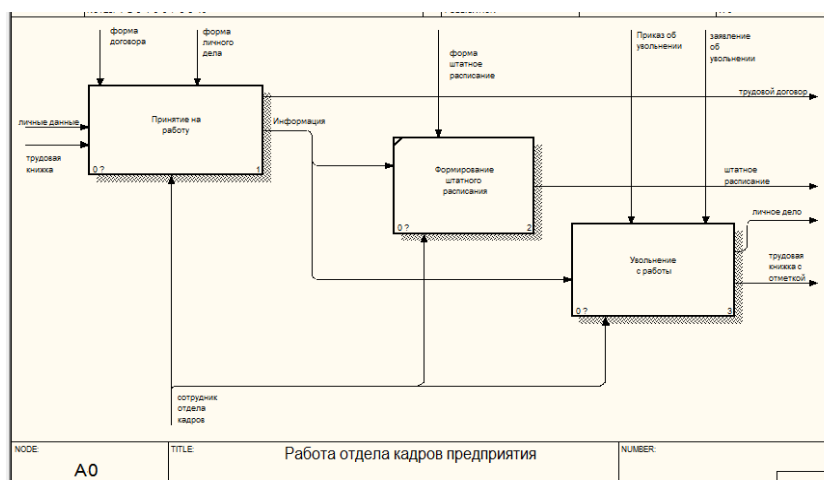


Рисунок 1.2 – Диаграмма декомпозиции бизнес-процесса деятельности отдела кадров филиала «ФБУЗ Центр гигиены и эпидемиологии» Новооскольского района Белгородской области по учету сотрудников

Основная функция разбивается на следующие работы:

- принятие на работу;
- формирование штатного расписания;
- увольнение с работы;

В свою очередь каждый блок-работа диаграммы декомпозиции может быть декомпозирован.

1.2 Экономическая сущность задачи

Если процессы кадрового учета в организации не автоматизированы, то это приводит к возникновению определенных сложностей:

- низкая скорость работы при обработке документов в отделе кадров;
- возможность возникновения ошибок;
- увеличение бумажного документооборота;
- сложность поиска необходимых данных по каждому сотруднику;
- трудности в персональном учете кадров на предприятии;

Автоматизация учета кадров является экономически целесообразной, так как снижает трудозатраты на обработку информации и позволяет оперативно выдавать требуемые данные, позволяет повысить производительность труда, а также уменьшить количество ошибок.

Основным экономическим преимуществом данной системы по сравнению с лучшими отечественными и зарубежными образцами и аналогами является ее цена, которая включает себестоимость разработки, создания и внедрения. Можно выделить следующие экономические преимущества:

- скорость работы программы позволит сформировать большое количество отчетов в единицу времени, чем это было бы сделано вручную;
- для работы с программой достаточно одного сотрудника;
- использование системы в организации, косвенно влияет на конечный результат через повышение качества работы.

Значит, остальной персонал можно задействовать на другие работы. В случае использования разрабатываемой системы, необходимость в администраторе, имеющие глубокие навыки администрирования отпадают, так как интерфейс предоставляет все возможности управления системой, в том числе неквалифицированным персоналом.

1.3 Обоснование необходимости и цели использования вычислительной техники для решения задачи

В настоящее время большая часть российских предприятий уже завершила автоматизацию основных подразделений и теперь вплотную подошла к необходимости внедрения новых технологий в службах персонала.

Современные системы автоматизации позволяют значительно облегчить труд кадровиков и упорядочить все учетные и расчетные процессы, связанные с персоналом. Благодаря возможностям быстрого поиска и доступа к необходимой информации о сотрудниках обеспечиваются следующие преимущества в текущей работе кадровых сотрудников:

- существенно экономятся временные затраты за счет избавления от бумажной волокиты и ускорения документооборота;
- повышается прозрачность учета, возможность контроля работы сотрудников кадровых служб;
- обеспечивается доступность и быстрота поиска документов;
- осуществляется автоматический контроль правильности составления и заполнения кадровой документации;
- появляется возможность эффективного перераспределения обязанностей сотрудников благодаря снижению рутинной обработки бумажных документов;
- гарантируется соответствие кадровой документации меняющимся требованиям российского законодательства, так как фирма разработчик отслеживает изменения и выпускает обновления программ;

- снижается трудоемкость представления регламентированной отчетности в государственные органы;
- производится подготовка данных для автоматизированного расчета заработной платы.

В результате внедрения автоматизированной системы кадрового документооборота кадровая служба решает две задачи.

Во-первых, кадровики получают ценный инструмент автоматизации рутинных задач, возможность формировать гибкие отчеты о работниках с различными условиями отбора и сортировки.

Во-вторых, в результате автоматизации рутинных процедур сотрудники кадровой службы больше времени могут посвятить задачам, нацеленным на повышение эффективности работы персонала, что особенно важно в современных условиях.

1.4 Постановка задачи

1.4.1 Цель и назначение автоматизированного варианта решения задачи

Среди задач автоматизации деятельности отдела кадров филиала ФБУЗ «Центр гигиены и эпидемиологии» Новооскольского района Белгородской области отдельно следует выделить задачу автоматизации ведения базы учета сотрудников.

Автоматизация базы данных сотрудников даст возможность быстрее и полнее реагировать на запросы руководства, производить сложные выборки и поиски по многим параметрам, что существенно превосходит сложившуюся в данный момент времени ситуацию, когда все данных хранятся в текстовых документах формата MicrosoftWord.

Целью ВКР является совершенствование документооборота за счет автоматизации учета персональных данных о сотрудниках филиала «ФБУЗ

Центр гигиены и эпидемиологии» Новооскольского района Белгородской области.

Разрабатываемая система должна решать задачу автоматизации работы инспектора ОК и выполнять следующие функции:

- ввод данных о сотрудниках предприятия, их занимаемой должности и образовании;
- формирование отчетов о сотрудниках;
- хранение информации о кадровом обеспечении предприятия в информационной базе системы;
- учет данных о лабораторном оборудовании, которое числится за сотрудником;
- сведения о родственниках сотрудников

Программа должна работать без прерываний при возникновении критических ошибок, даже тех, которые возникают по вине пользователя или из-за ошибок во входных данных. Соответственно, секции программы, в которых возможно возникновение таких ошибок должны обрабатываться в программе особым образом. В случае возникновения ошибки после выдачи соответствующего предупреждения программа должна продолжить свою работу.

Система должна обеспечить физическую и логическую целостность базы данных. При изменении или удалении данных программа должна поддерживать ссылочную целостность информации в базе данных.

1.4.2 Общая характеристика организации решения задачи на ЭВМ

Задача автоматизации работы ОК состоит из нескольких основных подзадач это:

- 1) Создание базы данных с удобным методом хранения.
- 2) Создание подсистем автоматизированной системы для различных нужд.

3) Создание путей ввода информации в базу данных:

– ввод информации в базу данных должен быть разделен для всех таблиц или логических сущностей базы;

– ввод информации в базу должен иметь удобный интерфейс.

4) Создание путей получения информации из базы данных.

Основным путем получения данных из базы данных должны являться предварительно сформированные формы и отчеты.

5) Организация законченной структуры учета сотрудников.

База данных АС должна соответствовать следующим требованиям:

1) Доступность – база данных должна быть легко доступна по сети для многопользовательской работы.

2) Отсутствие необходимости обслуживания со стороны специально обученного персонала – что увеличивает расходы на содержание АС и увеличивает время простоя в случае сбоев.

3) Легкая возможность резервного копирования и восстановления – отсутствие требований по организации сложной процедуры резервного копирования и восстановления связано с отсутствием квалифицированных специалистов и нежеланием нести дополнительные расходы на них.

4) Легкая возможность модернизации – простота и открытость АС является одним из основных требований для страховки компании от невозможности разработчика данной АС далее работать и обслуживать её.

АС должна предоставлять полные возможности ведения базы сотрудников.

Отсутствие высоких требований к аппаратным компонентам – для отсутствия высоких дополнительных затрат на модернизацию парка ЭВМ организации.

Задача автоматизации работы ОК в данном проекте делится на несколько подзадач, успешное решение которых приводит к успешному решению задачи автоматизации деятельности инспектора ОК в целом. Можно выделить следующие подсистемы:

- подсистема ведения списка сотрудников предприятия, их занимаемой должности и образования;
- подсистема ведения списка оборудования предприятия;
- подсистема ведения списка родственников сотрудника;
- подсистема ведения справочников;
- подсистема подготовки и вывода отчетной информации.

1.5 Анализ существующих разработок и обоснование выбора технологии проектирования

Современные технологии делают возможной настоящую революцию в управлении кадрами. Сотрудники и управленческий персонал получают доступ к всеобъемлющей системе данных по кадровым ресурсам предприятия, что повышает уровень и скорость принятия решений, а также автоматизирует и систематизирует рутинные процессы, позволяя сконцентрироваться на стратегических задачах. Именно поэтому растет интерес к информационным системам управления персоналом (ИСУП).

В разрезе информационных технологий ИСУП представляет собой набор определенного программного обеспечения (ПО) и технологий, позволяющих автоматизировать и совершенствовать бизнес процессы в таких областях, как: управление кадрами, расчет и выплата зарплаты, табельный учет и документооборот.

Данное ПО не только позволяет координировать действия различных департаментов, но и координировать работу различных каналов взаимодействия между персоналом и руководством: личное взаимодействие, телефон, Интернет. Кроме того, данное ПО дает работникам отдела персонала доступ к полной информации о работнике, необходимой для наилучшего планирования и контроля бюджетов по зарплате, обучению, командировкам и т.д.

В Таблице 1.1 приведена классификация ИСУП.

Таблица 1.1–Классификация ИСУП

Локальные	Средние интегрированные	Крупные интегрированные
1С:Зарплата и кадры БЭСТ. Зарплата АиТ:\ Управление персоналом ИНФИН. Зарплата Инфософт.Зарплата Клуб Капитал Oracle-Кадры	Robertson&BlumsCorporation. Управление персоналом и зарплата. БОСС-Кадровик Парус. Персонал Scala. Зарплата Монолит-Инфо. Кадры, зарплата Галактика. Персонал NavisionАхарта. Персонал Exact. Payroll PeopleSoft	SAP R/3. Персонал JD Edwards. Управление персоналом. Предствитель в РФ - Robertson&Blums OracleApplications: Управлениеперсоналом BaanHR&Payroll.

Начнем с крупных интегрированных систем, которые, как правило, имеют модуль – «Управление персоналом и зарплата» в составе ERP-системы (EnterpriseResourcePlanningsystem – система управления ресурсами предприятия) и отдельно от этой системы не продается. А так как стоимость таких проектов может составлять миллионы долларов, то такие системы интересны крупным предприятиям сверхприбыльных отраслей. Поэтому здесь, логичней сделать обзор ERP-систем, но это не входит в нашу задачу. Поэтому рассматривать мы будем только локальные и средние ИСУП.

На другом полюсе нашей классификации стоят локальные ИСУП, среди которых можно выделить «1С. Зарплата/кадры» - представляющее простой, коробочный вариант по внедрению и учетной системой по функциональности, но за счет практически монопольного положения компании на своем рынке занимающее доминирующее положение.

АиТ– «Управление персоналом» – где по сравнению с другими функциональность расширена, но поддержка клиентов оставляет желать лучшего. Другие ИСУП в этой категории распространены меньше и имеют

свои достоинства и недостатки. Выбор ИСУП зависит от отрасли, в которой работает компания, связей руководства фирм-разработчиков и множества других причин, на первом месте среди которых – цена.

Существуют две системы, удерживающие лидирующие позиции на рынке средних ИСУП, именно они нам наиболее интересны для сравнения: «RB HR&Payroll - Управление кадрами и Зарплата» - Робертсон и БлумсКорпорейшн и «БОСС-Кадровик» – Корпорация АйТи.

Остальные компании разработчики, как правило, продвигают и продают комплексные финансово-управленческие системы, либо ERP-системы, в состав которых входит модуль по управлению человеческими ресурсами и примеры внедрения отдельно таких модулей единичны. То есть, чтобы автоматизировать управление кадрами, клиенту необходимо купить всю ERP-систему, иначе инвестиции в информационные технологии будут неэффективны (так называемая – кусочная автоматизация).

RB HR&Payroll - функционально полный, стабильный, гибкий и легко управляемый продукт.

Это международное решение для предприятий различного уровня и сфер деятельности, позволяет автоматизировать и упорядочить типичные операции в сфере управления персоналом:

- управление штатным расписанием;
- прием, продвижение по службе, перевод между подразделениями, увольнение работников;
- обучение и аттестация персонала;
- поддержка разнообразных компенсационных схем;
- расчет зарплаты;
- управление бюджетами подразделений по зарплате, найму, обучению персонала и др.

Уникальность RB HumanResources&Payroll определяет наличие таких характеристик, как мощнейшие средства анализа информации; работа на различных платформах, ОС и СУБД; открытый доступ к данным

присовершенной защите информации; настраиваемость на законодательства различных стран; пользовательские процедуры; ведение архива справочной информации; удобный интерфейс со встроенными средствами анализа; многоэкранные справочники работников; настраиваемые интеллектуальные экранные формы; настраиваемость расчета заработной платы, выполнения расчетов и получения отчетов; двухуровневая настройка расчетов зарплаты и платежей предприятия (таблицы видов оплат и алгоритмы расчетов); расчет зарплаты с возможностью многократного пересчета и исправления ошибок; мощный встроенный генератор отчетов; поддержка документооборота; масштабируемость; параметризируемые средства переноса результатов расчета зарплаты в систему бухгалтерского учета.

Схожие наборы характеристик имеются и в "БОСС-Кадровик", но в отличие от него, RB HR&Payroll полностью интегрирован с CrystalReports(генератор бизнес отчетов); имеет интерфейс с Lotus R/5, возможен доступ к системе с помощью браузера MS IE (HTML – клиент); поддерживает несколько языков и многовалютную обработку данных, функциональность продукта позволяет в течение нескольких часов перейти с одной платформы на другую, при этом не требуется переобучение персонала, имеет интерфейсы с SunSystems, Scala, Concord, JD EdwardsOneWorld, SAP R/3, а также может быть интегрирован с финансовыми модулями любого другого программного обеспечения. А модуль «Табельный Учет» считается лучшим среди всех существующих систем.

Для того чтобы ориентировочно оценить, во что может обойтись компании автоматизация управления персоналом представим таблицу 1.2.

Таблица 1.2– Внедрение, соотношение затрат и стоимостные оценки

	Локальные системы	Средние интегрированные	Крупные интегрированные
Внедрение	Простое, коробочный вариант	Поэтапное или как коробочный вариант. От 1 до 6-ти месяцев.	Поэтапное, сложное. Более 9-12 месяцев.
Функциональная полнота	Учетные (по направлениям)	Комплексный учет и управление персоналом	Комплексный учет и управление, дополнительные функции.
Соотношение затрат: Лицензия \ внедрение \ оборудование	1\0,5-1\2	1\1-2\1	1\1-5\1
Ориентировочная стоимость	200 – 5000 USD	5 – 300 тысяч USD	От 500 тысяч USD и выше

Выводы:

Для малых предприятий наиболее эффективным решением по соотношению цена/качество является разработка системы своими силами.

Таким образом, подводя итоги можно сказать, что выбирая тех или иных разработчиков, следует учитывать, что ИСУП это лишь инструмент и успех его применения зависит от вас и работников отдела персонала.

2 Обоснование проектных решений

2.1 Обоснование проектных решений по техническому обеспечению

Техническое обеспечение (ТО) - комплекс технических средств, предназначенных для работы информационной системы, а также соответствующая документация на эти средства и технологические процессы

Комплекс технических средств составляют:

- компьютеры любых моделей;
- устройства сбора, накопления, обработки, передачи и вывода информации;
- устройства передачи данных и линий связи;
- оргтехника и устройства автоматического съема информации;
- эксплуатационные материалы и др.

Для функционирования проектируемой ИС необходимо следующее аппаратное обеспечение:

- персональные компьютеры для рабочего места инспектора ОК;
- локальная вычислительная сеть;
- принтер для печати документов.

В настоящее время у инспектора ОК имеется компьютер. Конфигурация компьютера удовлетворяет минимальным требованиям к системе (см. таблица 2.1). Компьютер включен в сеть, имеет доступ к сети Интернет и оснащен лазерным принтером.

Таблица 2.1 – Минимальные требования к аппаратному обеспечению

Характеристика	Значение, не менее
Частота процессора	1 ГГц
ОЗУ	1 Гб
Объем жесткого диска	300 Гб
Монитор	1240x1024

Для проектирования и разработки автоматизированной информационной системы «Отдел кадров Центра гигиены и эпидемиологии Белгородской области Новооскольского района» необходимо следующее техническое обеспечение:

Персональный компьютер:

- процессор Intel®Core™2Quad CPUQ6600 @2.40 GHz;
- оперативная память – 2,00 Гб;
- объем накопителя на жестких магнитных дисках – 300 Гб.

Периферийные устройства:

- мышь;
- клавиатура;
- лазерный принтер.

Программное обеспечение:

- операционная система Windows Vista Ultimate;
- Браузер Opera 11.62.

2.2 Обоснование проектных решений по информационному обеспечению

Информационное обеспечение (ИО) представляет собой совокупность проектных решений по объемам, размещению, формам организации информации, циркулирующей в системе. Оно включает в себя:

- совокупность показателей, справочных данных, классификаторов и кодификаторов информации;
- унифицированные системы документации, специально организованные для автоматического обслуживания;
- массивы информации на соответствующих носителях;
- персонал, обеспечивающий надежность хранения, своевременность и качество технологии обработки информации.

Требования к составу, структуре и способам организации данных в системе:

1) Состав данных информационной системы должен максимально полно отражать предметную область, т.е. в состав данных должны входить все необходимые реквизиты и показатели входной и выходной информации.

2) Информация в системе должна храниться в реляционной БД, представляющей собой совокупность нормализованных таблиц.

3) Информационный обмен между компонентами осуществляется за счет единого информационного пространства.

Требования по применению систем управления БД:

1) Проектируемая система должна разрабатываться с использованием СУБД Access.

Требуемые характеристика языка описания БД:

- наглядность;
- простота изучения;
- обеспечение требуемой степени независимости данных;
- встроенные средства защиты от несанкционированного доступа.

Требования к структуре сбора, обработки, передачи данных в системе и представлению данных:

1) Ввод информации в систему реализуется посредством ручного ввода с использованием экранных форм. Таким образом, должны быть разработаны экранные формы ввода, обеспечивающие минимальную трудоемкость ввода.

2) Система должна содержать необходимые средства поиска информации в таблицах БД. Вывод информации должен осуществляться в виде экранных форм, отчетов, таблиц.

Требования к защите данных от разрушений при авариях и сбоях в электропитании системы: для защиты накопленных за несколько месяцев данных от разрушений при авариях должно быть предусмотрено резервное копирование БД на отдельные, не связанные напрямую с системой, носители информации.

Требования к контролю, хранению, обновлению и восстановлению данных: в целях контроля и исправления данных вся информация из БД должна быть доступна для изменения вручную.

Принципы организации ИО:

- 1) непротиворечивость и достоверность данных;
- 2) маленький объем хранения;
- 3) актуальность данных;
- 4) простота и удобство эксплуатации;
- 5) гибкость (возможность дальнейшего расширения базы данных и адаптации к изменениям в предметной области).

2.3 Обоснование проектных решений по программному обеспечению

Программное обеспечение (ПО) включает совокупность программ, реализующих функции и задачи ИС и обеспечивающих устойчивую работу комплексов технических средств. В состав программного обеспечения входят общесистемные и специальные программы, а также инструктивно-методические материалы по применению средств программного обеспечения.

К общесистемному (общему) программному обеспечению относятся программы, рассчитанные на широкий круг пользователей и предназначенные для организации вычислительного процесса и выполнения часто встречающихся вариантов обработки информации. Они позволяют расширить функциональные возможности ЭВМ, автоматизировать планирование очередности вычислительных работ, а также автоматизировать работу программистов. Специальное (функциональное) программное обеспечение представляет собой совокупность программ, разрабатываемых при создании ИТ конкретного функционального назначения. Оно включает пакеты прикладных программ, осуществляющих организацию данных и их обработку при решении функциональных задач ИС.

При выборе комплекса технических средств для разработки системы, одним из важнейших критериев является выбор операционной системы. Операционная система управляет техническими средствами компьютера, поддерживает запуск и выполнение тех или иных программ и приложений, обеспечивает защиту данных, выполняет различные сервисные функции. Каждая программа пользуется средствами, предоставляемыми операционной системой. Таким образом, выбор операционной системы очень важен, так как он определяет набор программ и формат исполняемых файлов, а также их взаимодействие с операционной системой.

На компьютерах с архитектурой x86, используемых в качестве рабочих мест пользователей, чаще всего применяются следующие операционные системы:

- операционные системы семейства Windows от фирмы Microsoft,
- операционные системы Linux/BSD семейства (UNIX подобные) от различных фирм – разработчиков (RedHat, Debian, Novel, Mandrakesoft, Gentoo, Slackware, IBM, Oracle, NetBSD, OpenBSD, FreeBSD) [13].

Для разработки программного приложения автоматизированной обработки выбор той или иной операционной системы не повлияет на функциональность системы по причине того, что при реализации алгоритмов программного приложения не требуется использования каких-либо специфических функций операционной системы. Оба типа операционных систем позволяют разрабатывать программный продукт без потери его функциональности, по причине наличия программных сред (языков программирования) для обоих типов операционных систем.

Все из вышеперечисленных операционных систем содержат интерфейсы межсетевого взаимодействия, что позволяет использовать программное приложение в сети, для обмена данными и параллельной работы нескольких копий программного приложения с одними исходными данными. Оба типа операционных систем содержат в себе качественный интерфейс пользователя,

что также позволяет производить разработку программного приложения для любой из этих операционных систем.

В качестве операционной среды для разработки и применения программы была выбрана операционная система семейства Windows, в частности операционная система Windows7. Этот выбор обусловлен тем, что на сегодняшний день Windows7 является одной из наиболее распространенных операционных систем. Операционная система Windows7 обеспечивает стабильность работы, предоставляя пользователям возможность сосредоточиться на выполняемой работе.

Одним из важных требований, предъявляемых к проектированию информационных систем, эксплуатируемых совместно на технологической базе весьма ограниченных возможностей, является большая их однородность, позволяющая обеспечить совместимость, мобильность, переносимость.

В настоящее время большинство предприятий для хранения и обработки информации использует ресурс информационных технологий.

Ниже будут рассмотрены наиболее популярные программные средства, используемые для создания различных приложений.

CodeGear — дочернее подразделение компании Embarcadero Technologies. CodeGear занимается инструментами для разработки программного обеспечения, такими как Delphi, интегрированными средами разработки и сервером баз данных InterBase. Первоначально являлась дочерней фирмой компании Borland Software Corporation, создана 14 ноября 2006 года.

Продукты CodeGear:

- RAD Studio (включая Delphi, Delphi.NET, C++Builder, C#Builder);
- TurboDelphi;
- Delphifor PHP;
- Delphi;
- Delphi/400;
- JBuilder;
- InterBase;

- Turbo C#;
- Turbo C++;
- C++BuilderX;
- C#Builder;
- C++ Builder;
- 3rdRail;

CodeGear RAD Studio 2007 - новая версия среды быстрой разработки приложений (RAD).

Продукт предназначен для быстрой и эффективной разработки мощных Windows- и .NET-приложений баз данных.

CodeGear RAD Studio дает разработчикам гибкость в выборе операционной системы, которая наиболее полно удовлетворяет их требованиям при создании приложения для нескольких ОС Windows, включая программы для WindowsVista и Web-приложения.

Среди функций и возможностей решения CodeGear можно выделить следующие:

- расширенная поддержка Delphi для пользовательского интерфейса MicrosoftVistaAero, включая прикладные приложения, библиотеки визуальных компонентов (VCL) и поддержку VCL.NET для 3D эффектов и анимации пользовательского интерфейса, эффекта прозрачности, компонентов диалоговых окон для работы с файлами и задачам;

- поддержка языка Delphi для разработки в среде Microsoft .NET 2.0 (совместимо с .NET 3.0) и ASP.NET 2.0. ASP.NET представляет собой набор технологий в рамках .NET framework для создания Web-приложений и XML Web-сервисов;

- Delphi для .NET имеет поддержку параметризованных типов, позволяющая разработчикам .NET применять Delphi для создания и использования классов, используя любой тип структуры данных в качестве параметров.

Как и платформа Windows, аппаратное обеспечение компьютеров и сеть постоянно эволюционируют, так и продукты CodeGear для Delphi и C++ также совершенствуются, чтобы дать разработчикам возможность извлечь из перемен пользу, создавая новые возможности для независимых поставщиков программного обеспечения (ISV) и создателей клиент/сервер приложений на основе баз данных. CodeGear RAD Studio 2007 открывает новые возможности для разработчиков, которые интересуются разработкой высокопроизводительных Windows-приложений с высокими требованиями к GUI, прикладных программ, работающих с базами данных, Web-приложений для бизнеса, а также стремятся извлечь максимальную пользу из усовершенствований аппаратного обеспечения и операционной системы Windows.

Microsoft Visual Basic — средство разработки программного обеспечения, разрабатываемое корпорацией Microsoft и включающее язык программирования и среду разработки. Язык Visual Basic унаследовал дух, стиль и отчасти синтаксис своего предка — языка Бейсик, у которого есть немало диалектов. В то же время Visual Basic сочетает в себе процедуры и элементы объектно-ориентированных и компонентно-ориентированных языков программирования. Среда разработки VB включает инструменты для визуального конструирования пользовательского интерфейса.

Visual Basic считается хорошим средством быстрой разработки прототипов программы, для разработки приложений баз данных и вообще для компонентного способа создания программ, работающих под управлением операционных систем семейства MicrosoftWindows.

Visual Basic.NET не позволяет программировать по-старому, ибо по сути является совершенно другим языком, таким же, как и любой другой язык программирования для платформы .NET. Индивидуальность языка, так же как и его преимущества (простота, скорость создания программ, лёгкость использования готовых компонент) при использовании в среде .NET не имеют такого значения, как раньше — всё сосредоточено на возможностях самой

системы .NET, на её библиотеке классов. Поэтому нужно говорить о классическом, его диалектах Visual Basic for Applications (VBA) и Visual Basic Scripting Edition (VBScript) и о языке для платформы .NET — Visual Basic.NET

Достоинства:

- быстрота создания приложений с графическим интерфейсом для MS Windows;

- простой синтаксис, позволяющий очень быстро освоить язык;

- возможность компиляции как в машинный код, так и в Р-код (по выбору программиста). В режиме отладки программа всегда (вне зависимости от выбора) компилируется в Р-код, что позволяет приостанавливать выполнение программы, вносить значительные изменения в исходный код, а затем продолжать выполнение: полная перекомпиляция и перезапуск программы при этом не требуется;

- защита от ошибок, связанных с применением указателей и доступом к памяти. Этот аспект делает VisualBasic приложения более стабильными, но также является объектом критики;

- возможность использования большинства WinAPI функции для расширения функциональных возможностей приложения. Данный вопрос наиболее полно исследован Дэном Эпплманом, написавшим книгу «Visual Basic Programmer's Guide to the Win32 API»;

- Visual Basic выгодно отличается от других языков программирования своей простотой и наглядностью;

- Visual Basic динамично развивающийся язык;

- Visual Basic встроен в такие программы как Word, Excel и др. С его помощью можно управлять этими программами из других программ;

- Visual Basic – это объектно-ориентированный язык. Основой языка являются объекты. Например: окно, кнопка, поле со списком, с которыми работает программа.

Недостатки:

- поддержка операционных систем только семейства Windows и Mac OS X (Исключение — VB1 for DOS);

- отсутствие механизма наследования реализации объектов. Существующее в языке наследование позволяет наследовать только интерфейсы, но не их реализацию;

- требует установленную `msvbvmXX.dll` для работы программы;

- низкая скорость работы, обусловленная тем, что практически все встроенные функции языка реализованы через библиотеку времени исполнения, которая, в свою очередь, производит много «лишней» работы по проверке и/или преобразованию типов.

Поскольку эта среда выполняется на компьютере вместе с операционной средой Windows, то и создавать вы будете программы, называемые Windows-приложения или просто: приложения. Для создания приложения необходимо составить проект. Этим вы и будете заниматься в среде программирования VisualBasic.

Microsoft Office Access или просто Microsoft Access — реляционная СУБД корпорации Microsoft. Имеет широкий спектр функций, включая связанные запросы, связь с внешними таблицами и базами данных. Благодаря встроенному языку VBA, в самом Access можно писать приложения, работающие с базами данных.

Основные компоненты MS Access:

- построитель таблиц;

- построитель экранных форм;

- построитель SQL-запросов (язык SQL в MS Access не соответствует стандарту ANSI);

- построитель отчётов, выводимых на печать.

Возможности среды позволяют вызывать скрипты на языке VBA, поэтому MS Access позволяет разрабатывать приложения и БД практически «с нуля» или написать оболочку для внешней БД.

MS Access является файл-серверной СУБД и потому применима лишь к маленьким приложениям. Отсутствует ряд механизмов, необходимых в многопользовательских БД, таких, например, как триггеры.

Существенно расширяет возможности MS Access по написанию приложений механизм связи с различными внешними СУБД: "связанные таблицы" (связь с таблицей СУБД) и "запросы к серверу" (запрос на диалекте SQL, который "понимает" СУБД). Также MS Access позволяет строить полноценные клиент-серверные приложения на СУБД MS SQL Server. При этом имеется возможность совместить с присущей MS Access простотой инструменты для управления БД и средства разработки.

Access, при работе с базой данных, иначе взаимодействует с жёстким (или гибким) диском, нежели другие программы.

В других программах, файл-документ, при открытии, полностью загружается в оперативную память, и новая редакция этого файла (изменённый файл) целиком записывается на диск только при нажатии кнопки «сохранить».

В Access новая редакция содержимого изменённой ячейки таблицы записывается на диск (сохраняется) сразу, как только курсор клавиатуры будет помещён в другую ячейку (или новая редакция изменённой записи записывается на диск сразу, как только курсор клавиатуры будет поставлен в другую запись (в другую строку)?). Таким образом, если внезапно отключат электричество, то пропадёт только изменение той ячейки (или записи?), которую не успели покинуть.

Кроме того, в Access эта кнопка нужна в режиме «Конструктор» для сохранения изменений структуры объекта базы данных, сделанных в этом режиме.

Система управления базами данных Microsoft Access является одним из самых популярных приложений в семействе настольных СУБД. Все версии Access имеют в своем арсенале средства, значительно упрощающие ввод и обработку данных, поиск данных и предоставление информации в виде таблиц, графиков и отчетов. Начиная с версии Access 2000, появились также Web-

страницы доступа к данным, которые пользователь может просматривать с помощью программы Internet Explorer. Помимо этого, Access позволяет использовать электронные таблицы и таблицы из других настольных и серверных баз данных для хранения информации, необходимой приложению. Присоединив внешние таблицы, пользователь Access будет работать с базами данных в этих таблицах так, как если бы это были таблицы Access. При этом и другие пользователи могут продолжать работать с этими данными в той среде, в которой они были созданы.

Oracle (Oracle Corporation) — американская корпорация, крупнейший в мире разработчик программного обеспечения для организаций, крупный поставщик серверного оборудования.

Компания специализируется на выпуске систем управления базами данных, связующего программного обеспечения и бизнес-приложений (ERP- и CRM-систем, специализированных отраслевых приложений). Наиболее известный продукт компании — Oracle Database, который компания выпускает с момента своего основания. С 2008 года корпорация освоила выпуск интегрированных аппаратно-программных комплексов, а с 2009 года в результате поглощения Sun Microsystems стала производителем серверного оборудования, до этого компания выпускала исключительно программное обеспечение.

Oracle Corporation - одна из крупнейших американских компаний, разработчик систем управления базами данных, инструментов для разработки баз данных, а также ERP-систем.

Oracle предлагает оптимизированный, полностью интегрированный комплекс программного и аппаратного обеспечения, помогающего компаниям осваивать сложные задачи и нестандартные подходы. Продукцией Oracle пользуются более чем 380 000 клиентов из различных отраслей в более чем 145 странах мира, включая все 100 компаний из списка Fortune 100.

Поэтому аппаратные и программные средства Oracle предполагают совместную работу в вычислительном облаке и в центре обработки данных - от серверов и систем хранения до баз данных и промежуточного ПО.

Достоинства Oracle:

- высокая производительность, надежность, защита и гибкость;
- сниженная стоимость и сложность реализации ИТ-проектов и управления ИТ-инфраструктурой;
- лучшая бизнес-аналитика, высокая продуктивность;
- клиентам, которых интересуют модульные решения, открытая архитектура Oracle и широкий выбор настроек операционных систем предлагают уникальные преимущества качественных продуктов во всех частях ИТ-комплекса, позволяя построить оптимальную инфраструктуру для данного бизнеса;
- Oracle упрощает системы ИТ, помогая клиентам развивать инновации и увереннее браться за сложные задачи.

Для разработки автоматизированной информационной системы «Отдел кадров центра гигиены и эпидемиологии Белгородской области Новооскольского района» целесообразно использовать среду разработки CodeGear RAD Studio.

Для реализации хранения и обработки информации при работе с автоматизированной информационной системой «Отдел кадров центра гигиены и эпидемиологии Белгородской области Новооскольского района» целесообразно использовать СУБД Microsoft Access.

2.4 Обоснование проектных решений по технологическому обеспечению

От того насколько рационально будет спроектирован технологический процесс, настолько гарантировано будет снижение стоимостных, трудовых затрат.

Технологический процесс, как правило, состоит из нескольких этапов. Целью первого этапа является сбор, регистрация, передача данных для дальнейшей обработки. Результатом обычно является составление документа. Цель второго этапа - перенос данных на машинные носители и первоначальное формирование информационной базы. Третий этап включает операции накопления, сортировки, корректировки и обработки данных.

При выборе варианта технологического процесса требуется учитывать следующие требования:

- обеспечение достоверности обрабатываемой информации;
- решение задач в установленные сроки;
- обеспечение минимальных трудовых и стоимостных затрат на обработку данных;
- наличие возможности обработки данных на ЭВМ;
- возможность решения задачи в различных режимах.

Исходя из перечисленных выше требований целесообразно проектирование АИС, которое позволит децентрализовать процесс решения задачи и повысить производительность.

При обработке данных желательно использовать массивы нормативно-справочной информации. Это дает преимущества в скорости поиска, выбора, сортировки и т.д. При этом необходима возможность просмотра полученных результатов перед оформлением и передачей выходной информации. Очень актуальным становится вопрос выбора режима: пакетный или диалоговый.

Пакетный режим позволяет уменьшить вмешательство пользователя в процесс решения задачи и требует от него только выполнения операций по вводу и корректировке данных, но вместе с этим появляется вероятность полной загрузки ЭВМ, что не всегда удобно для пользователя.

Практика показывает, что использование АИС с применением методов построения модели на основе диалога обеспечивает более гибкую связь пользователя с ЭВМ.

Диалоговый режим имеет ряд преимуществ: удобен при работе с базой; обеспечение защиты при несанкционированном доступе; обеспечивает непосредственное участие пользователя в процессе решения задачи; управляемость процессом; быстрый доступ, поиск и выдача информации в любой момент времени, выбор различных режимов работы; осуществление быстрого перехода от одной операции к другой.

Существует несколько типов диалога: управляющие команды, запросы, меню, диалог на ограниченном естественном языке.

В данной работе будет использоваться метод меню с многоуровневой структурой.

2.5 Обоснование выбора программных средств

В результате проведенного анализа современных программных комплексов для разработки автоматизированной информационной системы «Отдел кадров Центра гигиены и эпидемиологии Белгородской области Новооскольского района» были выбраны:

1) Среда визуальной разработки CodeGear RAD Studio 2007/Delphi, интерфейс которой представлен в приложении А. Среда быстрой разработки приложений, в которой интегрированы средства моделирования разработки и развертывания приложений электронной коммерции и Web-сервисов.

2) СУБД Microsoft Access 2007, внешний вид которой представлен в приложении Б. Microsoft Access на сегодняшний день является одним из самых распространенных настольных приложений для работы с базами данных. Это связано с тем, что Access обладает очень широким диапазоном средств для ввода, анализа и представления данных. Эти средства являются не только простыми и удобными, но и высокопродуктивными, что обеспечивает высокую скорость разработки приложений.

Выводы:

Для создания автоматизированной информационной системы «Отдел кадров центра гигиены и эпидемиологии Белгородской области Новооскольского района» целесообразно использовать среду разработки CodeGear RAD Studio.

Для реализации хранения и обработки информации при работе с автоматизированной информационной системой «Отдел кадров центра гигиены и эпидемиологии Белгородской области Новооскольского района» целесообразно использовать СУБД Microsoft Access.

3 Проектная часть

3.1 Информационное обеспечение задачи

3.1.1 Информационная модель и ее описание

В фазе бизнес – моделирования и разработки требований бизнес-логика может описываться в виде:

- текста;
- концептуальных аналитических моделей предметной области;
- бизнес-правил;
- разнообразных алгоритмов;
- диаграмм деятельности;
- графов и диаграмм перехода состояний;
- моделей бизнес-процессов.

Для описания бизнес логики используем методологию структурного анализа и проектирования SADT.

Для проектирования информационной модели использовалось CASE-средство ERwin, в результате чего была получена модель сущность-связь. В ней определены все основные объекты (сущности) и связи, которые существуют между ними. Были выделены 6 сущностей: «Должность», «Сотрудник», «Отдел», «Образование», «Оборудование», «Родственники» (см. рисунок 3.1).

Следующим этапом проектирования инфологической модели является детализация атрибутов сущностей. На этом этапе также проверяется соответствие полученной модели нормальным формам. Каждой сущности приписываются:

- 1) Первичный ключ.
- 2) Внешние ключи соответственно типам связей.
- 3) Атрибуты.

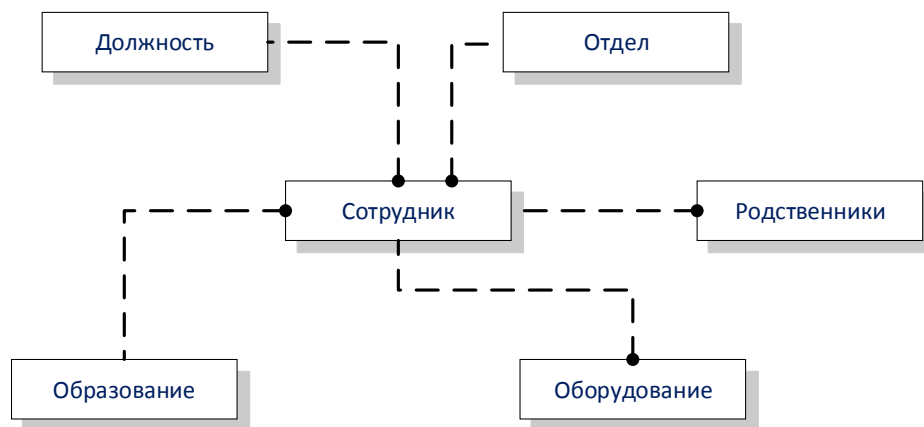


Рисунок 3.1 – Схема сущностей

Каждой сущности приписываются следующие атрибуты:

1) «Должность»:

- Код должности;
- Название должности;
- Выполняемые функции.

2) «Сотрудник»:

- Код сотрудника;
- Код должности;
- Код образования;
- Код отдела;
- Фамилия;
- Имя;
- Отчество;
- Специальность;
- Стаж работы;
- Стаж работы по специальности;
- Семейное положение;
- Дата рождения;
- Возраст;
- Пол;

- Национальность;
 - Адрес проживания;
 - Примечание.
- 3) «Отдел»:
- Код отдела;
 - Название отдела.
- 4) «Образование»:
- Код образования;
 - Вид образования;
 - Название учебного заведения;
 - Название отделения.
- 5) «Оборудование»:
- Код оборудования;
 - Наименование оборудования;
 - Изготовитель;
 - Дата эксплуатации;
 - Технические характеристики;
 - Документация;
 - Код сотрудника.
- б) «Родственники»:
- Код родственника;
 - Код сотрудника;
 - Фамилия;
 - Имя;
 - Отчество;
 - Дата рождения;
 - Степень родства;
 - Адрес родственника.

Таким образом, получаем инфологическую модель данных, представленную на рисунке 3.2.

Следующим этапом проектирования базы данных является даталогическое (логическое) проектирование базы данных.

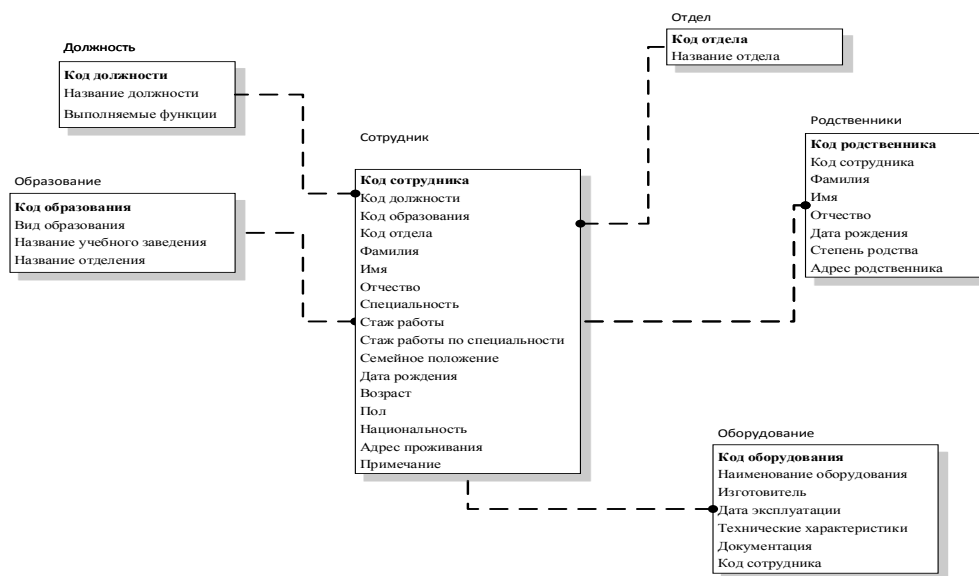


Рисунок 3.2 – Инфологическая модель

Даталогическая модель строится на основании инфологической модели и выполняется на языке описания данных конкретной СУБД. Каждому полю таблицы назначается имя, тип и размер. Полученная даталогическая модель в СУБД Access представлена на рисунке 3.3.

Необходимость построения и анализа схемы информационных потоков предметной области определяется сложностью и разнообразием существующего информационного обмена. Основной целью данного этапа проектирования является выделение внешних сущностей, осуществляющих информационный обмен с предметной областью, определение маршрутов и содержания информационных потоков, выделение и описание существующих хранилищ данных. Дополнительной целью этапа является более глубокое и системное изучение предметной области.

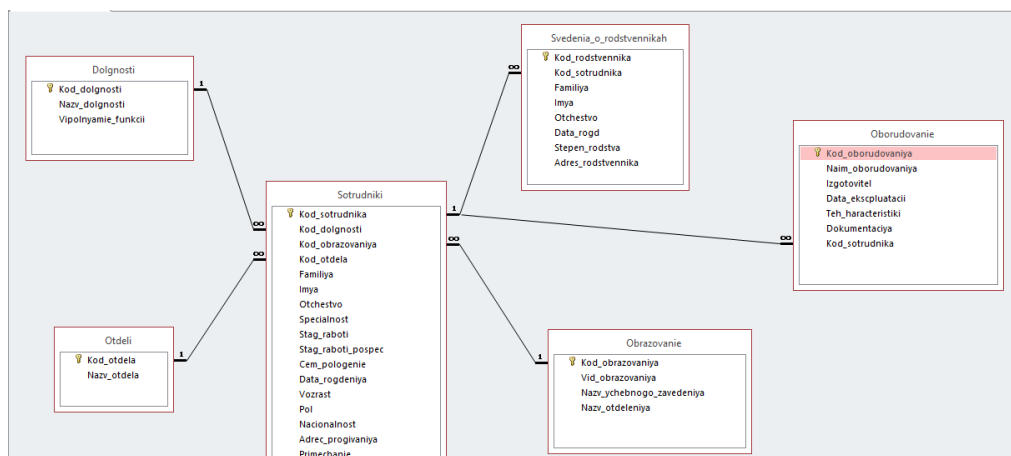


Рисунок 3.3 – Дatalogическая модель

Для построения и анализа схемы информационных потоков было использовано CASE-средство AllFusionProcessModeler версии 7.0 и, в частности, блок моделирования, поддерживающий нотацию DataFlowDiagram (DFD). Указанный блок позволяет реализовать задачу данного этапа проектирования посредством использования следующих объектов:

- Работы. Обозначают именованные процессы, функции или задачи, которые происходят в течение определенного времени и имеют распознаваемые результаты. Работы изображаются в виде скругленных прямоугольников.

- Информационные потоки. Отображают потоки информации как между работами в рамках предметной области, так и между работами и внешними сущностями или хранилищами данных.

- Внешние сущности. Представляют внешние объекты по отношению к предметной области, осуществляющие информационный обмен с внутренними объектами.

- Хранилища данных. Символизируют некоторые абстрактные объекты, предназначенные для аккумуляции информации. Каждому хранилищу на схеме соответствуют реальные объекты предметной области различной природы.

3.1.2 Используемые классификаторы и системы кодирования

Зарегистрированные общероссийские и международные классификаторы в системе не используются.

В системе используются следующие системные классификаторы:

- справочник должностей;
- справочник отделов;
- справочник образования.

Исходя из задач, поставленных перед нами при разработке данной системы из структуры информации и требований к системе, применение дополнительных классификаторов выглядит необоснованным и только перегрузит систему если они будут включены.

3.1.3 Характеристика первичных документов с нормативно-справочной и входной оперативной информацией

Под входной информацией понимается вся информация, необходимая для решения задачи и расположенная на различных носителях: первичных документах, машинных носителях, в памяти персонального компьютера. С этой целью составляется перечень входной информации и состав реквизитов каждого вида входной информации, расположение реквизитов входной информации, описание полей (реквизитов) входных документов.

При определении перечня входной информации описываются вид информации (текущая переменная, нормативно-справочная), источники информации, специфика сбора, хранения информации, способы поступления, а также объемно-временные характеристики и способы контроля.

Состав реквизитов входной информации зависит от особенностей входной информации. Он должен быть необходимым и достаточным для организации дальнейшей обработки. Расположение реквизитов осуществляется в соответствии с существующими правилами ее проектирования. Описание

полей (реквизитов) выполняется по отношению ко всем видам входной информации и осуществляется аналогично подобной операции для выходной информации.

В процессе возникновения и установления трудовых отношений в кадровой службе образуется множество документов под обобщающим названием «кадровая документация». Весь комплекс этих документов можно систематизировать по различным признакам.

По целевой принадлежности выделяют две больших группы кадровых документов:

1) Документы по учету личного состава работников, к которым относятся приказы о приеме на работу, переводе на другую работу, предоставлении отпуска, увольнении, личная карточка работника и другие. Основная часть документов по личному составу вошла в состав унифицированных форм первичной учетной документации по учету труда и его оплаты.

2) Вторую группу составляют документы, связанные с осуществлением функций по управлению персоналом и организацией труда (правила внутреннего трудового распорядка, положение о структурном подразделении, должностные инструкции, структура и штатная численность, штатное расписание). В «Общероссийском классификаторе управленческой документации» ОК 011-93, утвержденном Постановлением Госстандарта Российской Федерации от 30 декабря 1993 года №299 эти документы получили название «документация по организационно-нормативному регулированию деятельности организации, предприятия».

Разработчики вышеуказанного классификатора применяют и другой принцип систематизации кадровой документации, а именно по основным кадровым процедурам различают следующие виды кадровых документов:

- 1) Документация по приему на работу:
 - заявление о приеме на работу;
 - контракт о назначении на должность;
 - приказ о приеме на работу;

- протокол общего собрания трудового коллектива о приеме на работу.

2) Документация по переводу на другую работу:

- заявление о переводе на другую работу;
- представление о переводе на другую работу;
- приказ о переводе на другую работу.

3) Документация по увольнению с работы:

- заявление об увольнении;
- приказ об увольнении;
- протокол общего собрания трудового коллектива об увольнении.

4) Документация по оформлению отпусков:

- график отпусков;
- заявление о предоставлении отпуска;
- приказ о предоставлении отпуска.

5) Документация по оформлению поощрений:

- представление о поощрении;
- приказ о поощрении;
- протокол общего собрания трудового коллектива о поощрении.

6) Документация по оформлению дисциплинарных взысканий:

- докладная записка о нарушении трудовой дисциплины;
- объяснительная записка о нарушении трудовой дисциплины;
- приказ о наложении дисциплинарного взыскания;
- протокол общего собрания трудового коллектива о наложении дисциплинарного взыскания.

Исходя из основных управленческих задач, комплекс кадровых документов можно также представить в виде системы, включающей в себя следующие, связанные единством происхождения и различающиеся по функциональному назначению группы (подсистемы) документов:

1) Организационно-правовая документация:

- правила внутреннего трудового распорядка;

- штатное расписание;
- положение о структурном подразделении;
- график отпусков;
- должностные инструкции.

2) Персональная документация:

- трудовая книжка;
- личное дело;
- личная карточка работника.

3) Договорная документация:

- коллективный договор (соглашение);
- трудовой договор.

4) Распорядительная документация:

- приказы по личному составу.

5) Учетная кадровая документация:

- учетные формы (книги, журналы и другое) кадровой документации;
- документация по учету рабочего времени и расчетов с персоналом.

6) Информационно-справочная документация.

Наиболее многочисленную по видам и разновидностям группу кадровых документов составляет организационно-распорядительная документация, охватывающая вопросы организационно-правового обеспечения деятельности кадровой службы и ее сотрудников, документирования движения кадров, аттестации персонала, соблюдения трудовой дисциплины и многие другие.

3.1.4 Характеристика базы данных

Детализированные таблицы базы данных «SES.accdb»

Таблица «Sotrudniki» содержит сведения о сотрудниках.

Таблица «Dolgnosti» содержит сведения о должностях сотрудников организации.

Таблица «Oborudovanie» содержит сведения об оборудовании организации.

Таблица «Obrazovanie» содержит сведения об оборудовании организации.

Таблица «Otdeli» содержит сведения об отделах организации.

Таблица «Rodstveniki» содержит сведения о родственниках сотрудников организации.

Перечень таблиц, в которых показана их структура приведена в приложении Г.

3.1.5 Характеристика результатной информации

Отчетность организации служит основным источником информации о ее деятельности, так как отчеты собирают, накапливают и обрабатывают экономически существенную информацию об операциях и результатах деятельности. Таким образом, отчетность служит инструментом планирования и контроля достижения экономических целей предприятия.

На законодательном уровне практически во всех странах действует норма обязательности составления отчетности, чтобы обеспечить многочисленных пользователей нужной им информацией о том или ином субъекте рынка (организации).

Для внутренних пользователей отчетность является важным показателем оперативного управления.

На базе входной информации по кадровому учету система позволяет сформировать самые разнообразные показатели, которые составляют содержание форм отчетности, а также показатели, предназначенные для использования в других модулях.

Основными выходными отчетными формами разрабатываемой системы, наиболее часто используемыми специалистами по кадрам филиала ФБУЗ

"Центр гигиены и эпидемиологии» Новооскольского района Белгородской области, являются следующие формы:

- список сотрудников;
- список должностей сотрудников;
- личный листок сотрудника;
- список оборудования, относящегося к сотруднику предприятия;
- список отделов организации;
- список родственников сотрудника;
- сводный отчет по специальностям, образованию, стажу и семейному положению сотрудников;
- список образования сотрудников.

3.2 Программное обеспечение задачи

3.2.1 Общие положения

Автоматизированная информационная система «Отдел кадров Центра гигиены и эпидемиологии Белгородской области Новооскольского района» предназначена для автоматизации обработки персональных данных о сотрудниках организации, их должностях и образования.

Автоматизированная информационная система создавалась по заявке филиала ФБУЗ «Центр гигиены и эпидемиологии Новооскольского района Белгородской области».

Функциональные возможности данной автоматизированной системы:

- запуск приложения;
- добавление новых данных (ввод новой информации в базу);
- редактирование данных (позволяет изменить информацию в уже имеющихся данных);
- удаление данных (позволяет удалить информацию);

- фильтрация данных по определенному параметру (отбор данных по заданному условию);
- поиск данных (поиск информации, содержащей в базе по запросу пользователя);
- сортировка данных (позволяет упорядочивать данные по определенному параметру);
- формирование отчетов(позволяет выводить данные на экран и печать).

Разработанный продукт включает следующие компоненты:

- компонент представления данных;
- прикладной компонент;
- компонент управления ресурсом.

Приложение-клиентчерез ODBC передает запрос к базе данных, а обратно получает требуемые данные. Схема доступа к базе данных представлена в соответствии с рисунком 3.5.

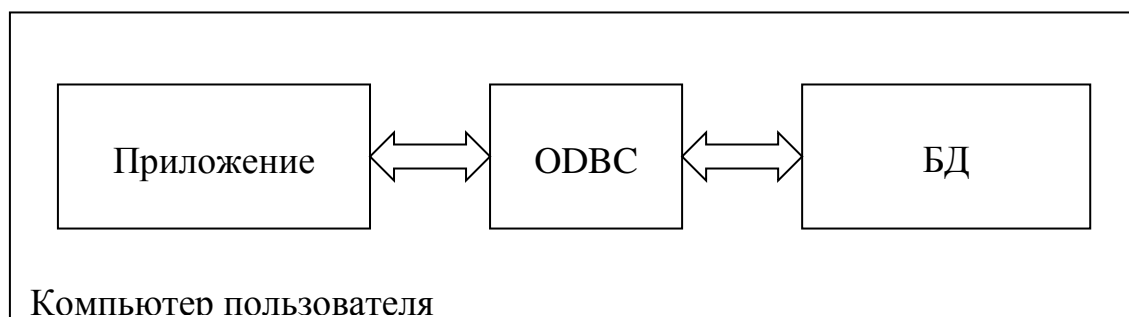


Рисунок 3.5–Схема доступа к базе данных

Open Database Connectivity (ODBC) является интерфейсом для доступа к данным в гетерогенной среде реляционных и нереляционной базы данных системы управления. В зависимости от уровня интерфейса вызова Спецификация SQL Access Group, ODBC предоставляет открытые, нейтральный поставщика способ доступа к данным, хранятся в различных специализированных персонального компьютера миникомпьютер и баз данных для больших ЭВМ.

Создавать пользовательский интерфейс программного продукта и различные приложения для базы данных целесообразно в инструментальных средах быстрой разработки приложений. Проектировать структуру базы данных, создавать таблицы и схемы данных необходимо с использованием систем управления базами данных. Шаблоны для формирования отчетной информации необходимо разрабатывать в инструментальных средах обработки языка разметки гипертекста.

3.2.2 Структурная схема пакета

Дерево иерархии функций системы представлена на рисунке 3.6.

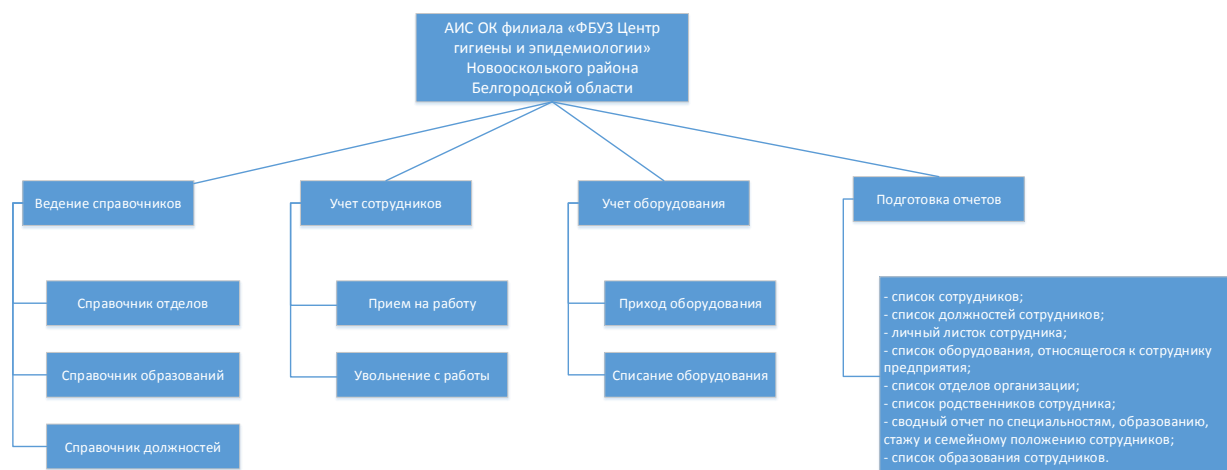


Рисунок 3.6–Дерево иерархии функции системы

В данном проекте можно выделить следующие категории функций: функции ввода информации в справочники, функции ввода данных о сотрудниках и оборудовании и функции получения аналитических отчетов. Каждая из этих категория относится соответствующей подсистеме разрабатываемого приложения: подсистеме ведения справочников, подсистеме ввода документов и подсистеме получения аналитических отчетов.

Приложение обеспечивает удобный и понятный диалог пользователя с программой для вызова всех функций. Доступ пользователя к функциям

производится через главное меню главной формы приложения и через кнопки панели быстрого запуска, дублирующие соответствующие пункты меню. Главная форма приложения кроме предоставления основного интерфейса доступа ко всем функциям приложения выполняет также роль контейнера для всех остальных экранных форм.

Первая группа экранных форм предоставляет пользователю доступ к подсистеме ведения справочников системы. Эта группа экранных форм вызывается из главного меню программы «Справочники». Каждый подпункт этого пункта меню соответствует вызову экранной формы, которая позволяет редактировать соответствующий справочник.

Все экранные формы просмотра и редактирования справочников похожи друг на друга и позволяют выполнять функции просмотра списка справочника, ввода новых данных в справочник, редактирования существующих данных и удаления имеющихся данных. Для ввода новых и редактирования существующих данных справочников вызываются формы редактирования соответствующих справочников. Эти же формы служат и для просмотра подробной информации записи справочника, так как в списке справочника представлены только основные данные.

Состав объектов, обеспечивающих функциональность клиентского приложения приведен в приложении В.

3.2.3 Описание программных модулей

Модульная структура клиентского приложения автоматизированной информационной системы «Отдел кадров Центра гигиены и эпидемиологии Белгородской области Новооскольского района» представлена в приложении Д.

Назначение модулей приложения:

- модуль FGlavnya– главный модуль программы;
- модуль FSotrudniki– модуль управления данными о сотрудниках организации;

- модуль FDolgnosti – модуль управления конкретными данными о должности сотрудника;
- модуль FOtdeli – модуль управления конкретными данными о названии отдела, в котором работает сотрудник;
- модуль FOborud – модуль управления конкретными данными об оборудовании, закрепленное к конкретному сотруднику;
- модуль FObrazov – модуль управления конкретными данными об образовании сотрудника;
- модуль FRodstven – модуль управления конкретными данными о родственниках сотрудника;
- модуль FSotrDob – модуль, обеспечивающий ввод нового сотрудника;
- модуль FSotrRed – – модуль редактирования информации о текущем сотруднике;
- модуль FDolgDob – модуль, обеспечивающий ввод новой должности;
- модуль FDolgRed– модуль редактирования информации о текущей должности;
- модуль FOtdDob – модуль, обеспечивающий ввод нового отдела;
- модуль FOtdRed – модуль редактирования информации о текущем отделе;
- модуль FOboruDob – модуль, обеспечивающий ввод нового оборудования;
- модуль FOborudRed – модуль редактирования информации о текущем оборудовании;
- модуль FObrazovDob – модуль, обеспечивающий ввод нового образования;
- модуль FObrazovRed – модуль редактирования информации о текущем образовании;

- модуль FRodstvenDob – модуль, обеспечивающий ввод нового родственника;
 - модуль FRodstvenRed – модуль редактирования информации о текущем родственнике;
 - модуль FOtchot – модуль формирования отчетной документации;
- Полный листинг программы представлен в приложении Е.

3.3 Технологическое обеспечение задачи

3.3.1 Организация технологии сбора, передачи, обработки и выдачи информации

При подробном анализе и изучении данной предметной области и диаграммы AS-IS можно прийти к выводу, что информационная система значительно упростит работу сотрудников отдела кадров.

С созданием данной системы функции учета сотрудников и закрепленного за ними оборудования автоматизируются и значительно упрощаются, поэтому уменьшается вероятность случайных ошибок, возникающих в силу влияния человеческого фактора.

Необходимо также отметить, что отдел кадров подразумевает работу с большими объемами данных и данная система призвана не только исключить случайные ошибки, но и уменьшить временные затраты на обработку, поиск данных, выполнение функций, составление отчетности.

Проведенный анализ позволил разработать модель TO-BE, контекстная диаграмма которой представлена на рисунке 3.8.

Таким образом, вся работа сотрудника отдела кадров по учету сотрудников с внедрением информационной системы, сводится к минимуму. Кроме того, к существующим в отделе отчетным формам, которые с внедрением системы будут формироваться и печататься из ИС, добавляются две новых, согласно пожеланиям работников ОК.

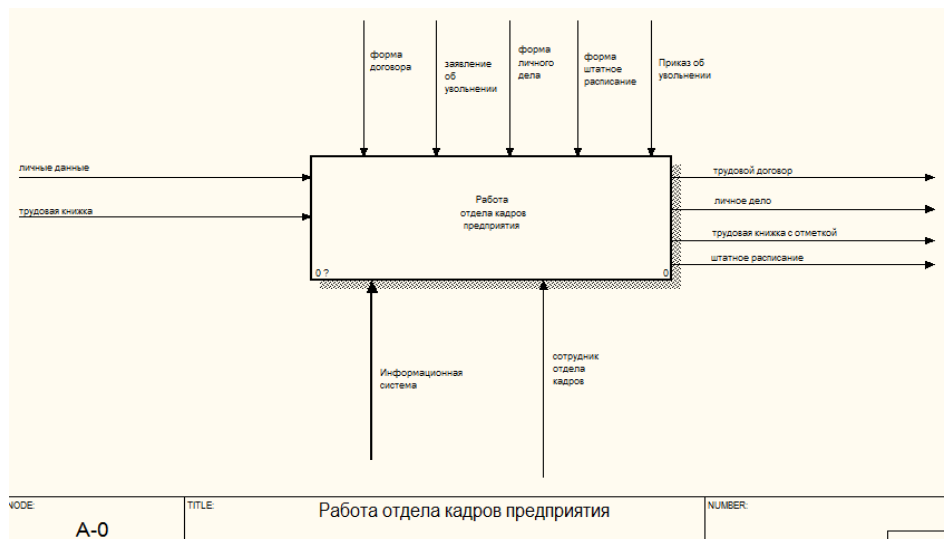


Рисунок 3.8 – Контекстная диаграмма ТО-ВЕ деятельности отдела кадров ФБУЗ «Центр гигиены и эпидемиологии» Новооскольского района Белгородской области

Декомпозиция контекстной диаграммы представлена на рисунке 3.9.

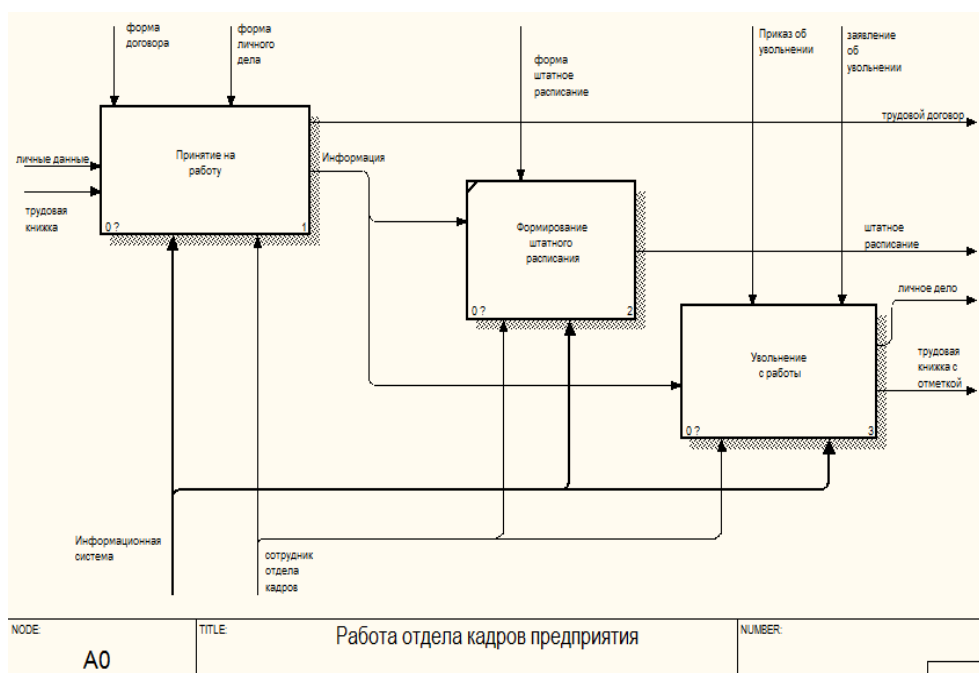


Рисунок 3.8 – Диаграмма декомпозиции ТО-ВЕ деятельности отдела кадров ФБУЗ «Центр гигиены и эпидемиологии» Новооскольского района Белгородской области

3.3.2 Схема технологического процесса сбора, передачи, обработки и выдачи информации

Цель разработки предложенной системы - интегрировать в одном программном продукте учет кадров и формирования на их основе соответствующих документов.

Цель использования данной системы - желание повысить эффективность работы отделов кадров за счет сокращения времени затрачиваемого на делопроизводство.

В соответствии с целью создания системы определяются и критерии создания автоматизированной системы. Она должна с одной стороны обладать алгоритмическим аппаратом, позволяющим выполнять поставленную задачу, с другой стороны не ограничивать пользователя системы в возможной свободе выбора формирования отчетов.

Система по своему предназначению имеет следующие основное ограничение: система предназначена для автоматизации делопроизводства в отделах кадрах.

Критерием эффективности работы системы является сокращение времени на формирование разных документов.

Система изначально создается с учетом русскоязычных пользователей. Такой подход более точно указывает на направления в реализации и совершенствовании алгоритмов и методов, используемых в системе.

Разработанная система позволяет накапливать и обрабатывать поступающие данные для дальнейшего использования, как для формирования разных отчетов, так и для автоматизации делопроизводства.

Данная система может применяться для автоматизации деятельности отделов кадров.

Сами конечные данные хранятся в БД, которая может свободно переноситься для использования в различные организации.

Сбор информации осуществляется ручным способом. Справочники вводятся пользователем.

При разработке системы, с одной стороны, решается вопрос о том, какие сведения и для каких целей будут содержаться в системе, с другой – как соответствующие данные будут организованы в памяти ЭВМ, как они будут поддерживаться, и обрабатываться при эксплуатации системы.

Разрабатываемая автоматизированная система обеспечивает ввод, систематизацию, хранение, обработку, поиск, выдачу на печать информации по запросу пользователя.

Цель автоматизированного варианта решения задачи – достижение прямого и косвенного экономических эффектов.

Прямой эффект – сокращение трудоемкости и стоимостных затрат при обработке кадрового учета сотрудниками ОК организации.

Косвенный эффект автоматизации деятельности инспектора ОК сводится к повышению оперативности составления документов; структуризации информации по сотрудникам предприятия в рамках БД; исключению задержки необходимой информации; повышению достоверности результатов; исключению несоответствия данных, используемых в различных отчетах; исключению дублирования потоков информации; снижению трудоемкости обработки информации; усовершенствованию процессов сбора, передачи, обработки, хранения, защиты целостности информации, выдачи результатов расчетов конечному пользователю.

Таким образом, автоматизация технологического процесса сбора, передачи, обработки и выдачи информации позволит:

- улучшить качество работы инспектора ОК;
- улучшить время подготовки отчетов;
- обеспечить представление документации в электронном виде, что позволит более эффективно использовать время данного сотрудника ОК.

3.4 Описание контрольного примера реализации проекта

Тестирование автоматизированных информационных систем представляют собой процесс проверки выполнения заданных функций системы, определения и проверки соответствия требованиям как количественных, так и качественных характеристик системы, выявления и устранения недостатков в действиях системы, в разработанной документации.

Объектом испытаний является автоматизированная информационная система «Отдел кадров центра гигиены и эпидемиологии Белгородской области Новооскольского района». Область применения программы – ФБУЗ «Центр гигиены и эпидемиологии». Обозначение программы «SES.exe».

Цель испытаний - проверка работоспособности автоматизированной информационной системы «Отдел кадров Центра гигиены и эпидемиологии Белгородской области Новооскольского района».

Требования к программе - проверка работоспособности клиентского приложения при выполнении нижеуказанных функций:

- запуск приложения;
- добавление новых данных (ввод новой информации в базу);
- редактирование данных (позволяет изменить информацию в уже имеющихся данных);
- удаление данных (позволяет удалить информацию);
- фильтрация данных по определенному параметру (отбор данных по заданному условию);
- поиск данных (поиск информации, содержащей в базе по запросу пользователя);
- сортировка данных (позволяет упорядочивать данные по определенному параметру);
- формирование отчетов(позволяет выводить данные на экран и печать).

Технические средства, используемые во время проведения испытаний:

Персональный компьютер:

- процессор Intel®Core™2Quad CPUQ6600 @2.40 GHz;
- оперативная память – 2,00 Гб;
- объем накопителя на жестких магнитных дисках– 300 Гб.

Периферийные устройства:

- мышь;
- клавиатура;
- лазерный принтер.

Программное обеспечение:

- операционная система Windows Vista Ultimate;
- программа для создания баз данных –Microsoft Office Access 2007;
- браузер Opera 11.62;
- программа для разработки интерфейса и приложений –CodeGear RAD Studio 2007/Delphi.

Методы испытаний представлены в Приложении Ж.

Для того чтоб правильно настроить программу, нужно выполнить следующие действия:

Меню Пуск → Настройки → Панель управления → Администрирование - Источники данных (ODBC). В результате откроется окно, представленное в соответствии с рисунком 3.9.

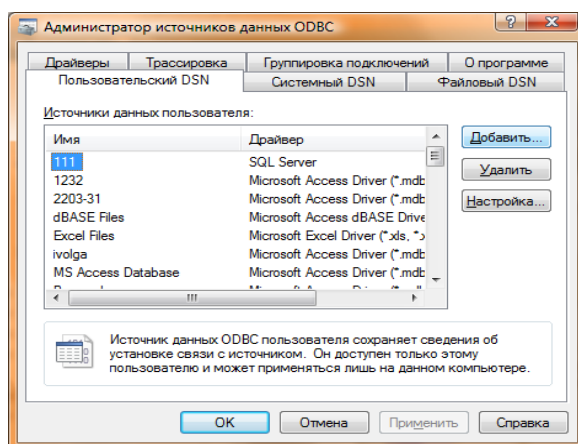


Рисунок 3.9–Администратор источников данных ODBC

В появившемся окне необходимо нажать кнопку «Добавить», для перехода в окно администратора источников ODBC, представленного в соответствии с рисунком 3.10.

Далее в раскрывшемся списке нужно выбрать драйвер, для которого задается источник, и нажать кнопку «Готово».

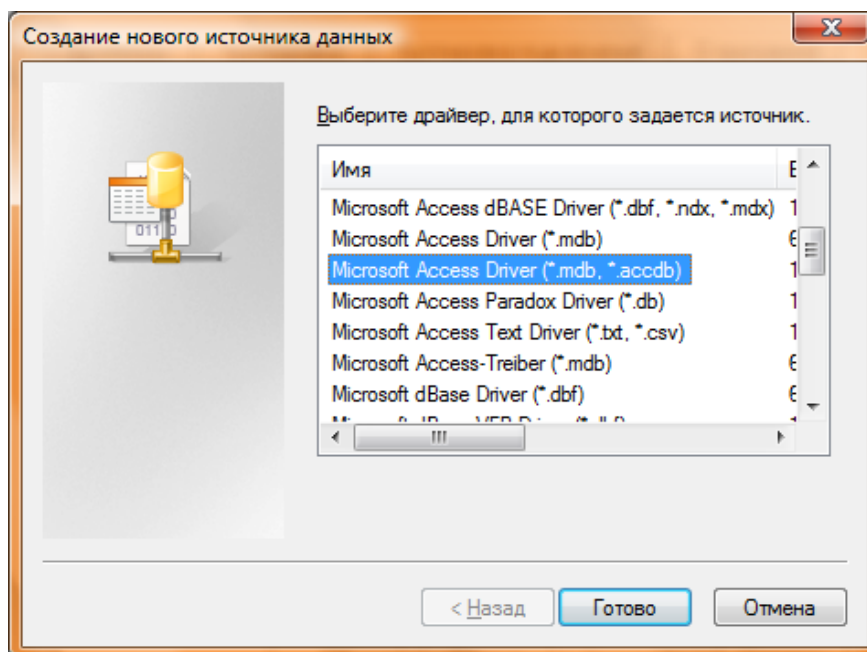


Рисунок 3.10–Создание нового источника данных

После этого в окне Установка драйвера ODBC для Microsoft Access произвести регистрацию базы данных, для чего необходимо задать имя источника данных и нажать кнопку «Выбрать». Выполнение этих действий представлено в соответствии с рисунком 3.11.

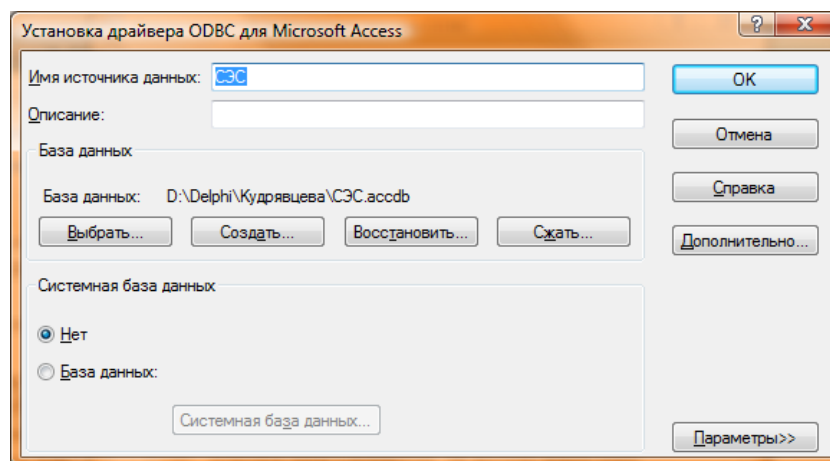


Рисунок 3.11 –Установка драйвера ODBC для Microsoft Access

Затем необходимо выбрать базу данных, а после нажать кнопку «Ок». Выполнение этого действия представлено в соответствии с рисунком 3.12.

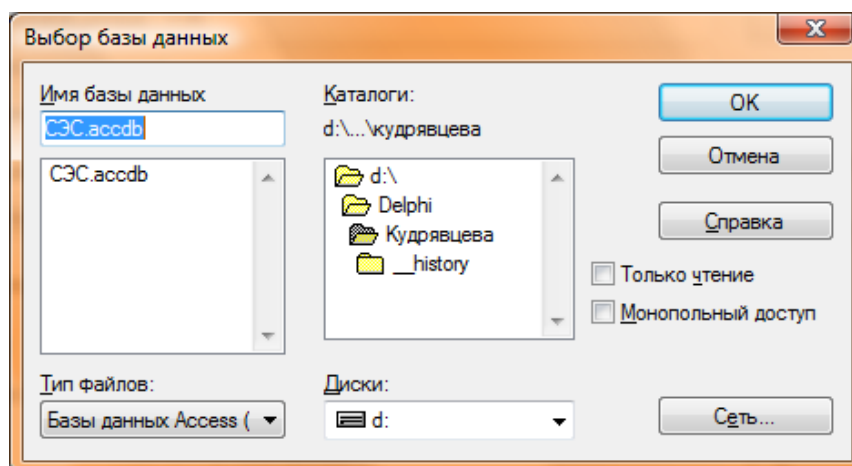


Рисунок 3.12 –Выбор базы данных

Тестирование автоматизированных информационных систем представляют собой процесс проверки выполнения заданных функций системы, определения и проверки соответствия требованиям как количественных, так и качественных характеристик системы, выявления и устранения недостатков в действиях системы, в разработанной документации. В ходе тестирования была проведена проверка работоспособности клиентского приложения при выполнении вышеуказанных функций.

Дополнительных возможностей разработанная автоматизированная система не имеет.

Сообщение системному программисту, о невозможности установления связи с базой данных при запуске автоматизированной информационной системы представлено в соответствии с рисунком 3.13.

Для устранения этой ошибки нужно выполнить настройку системы.

Чтобы запустить программу необходимо в папке «СЭС» двойным щелчком мыши кликнуть по файлу SES.exe, откроется окно автоматизированной информационной системы ««Отдел кадров Центра гигиены и эпидемиологии Белгородской области Новооскольского района», представленное в соответствии с рисунком 3.14.

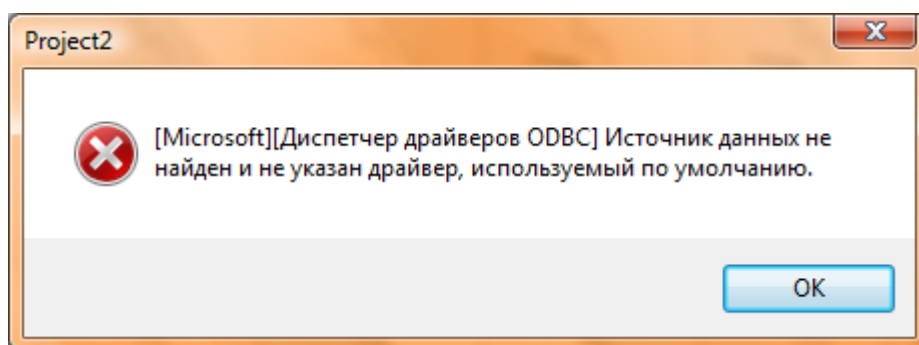


Рисунок 3.13 –Сообщение о невозможности установления связи с базой данных

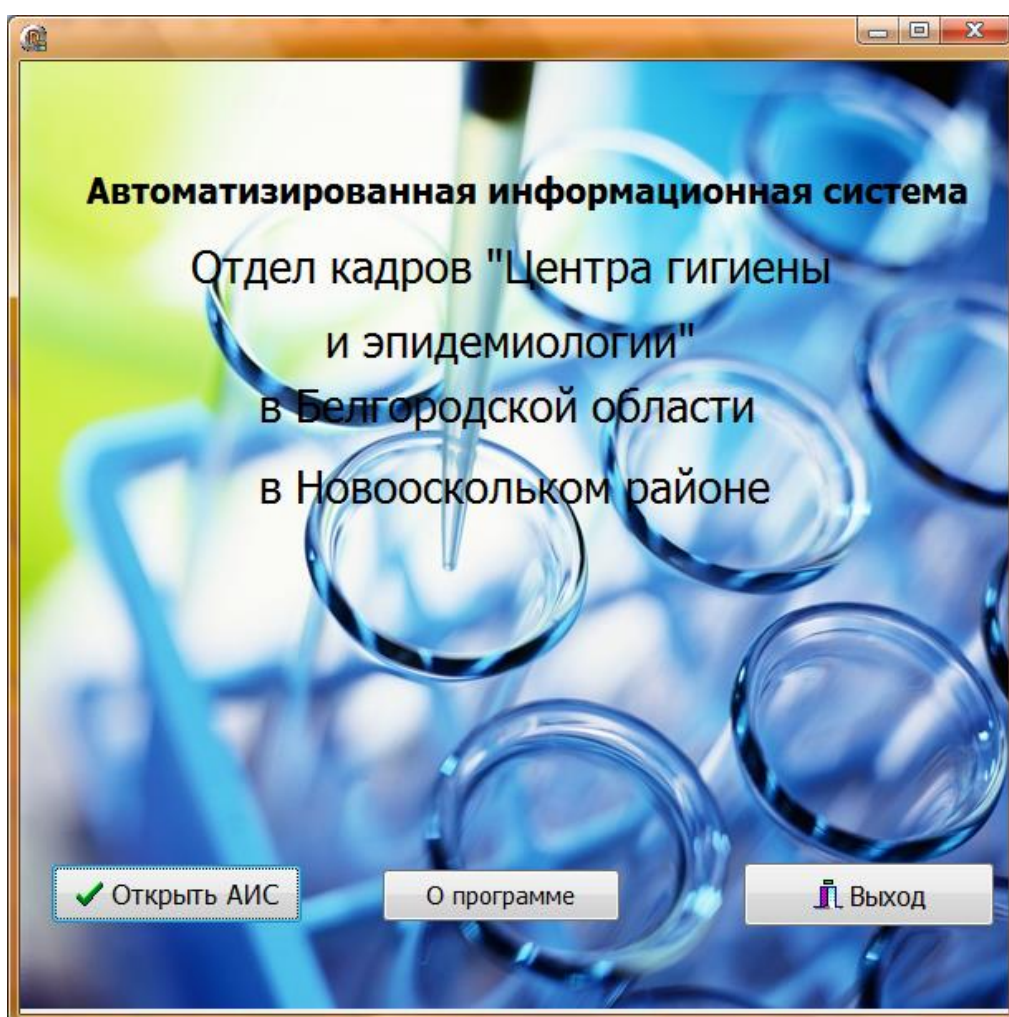


Рисунок 3.14 –Внешний вид АИС

Для открытия базы данных необходимо активировать кнопку «Открыть АИС». Откроется окно для ввода логина и пароля, представленное в соответствии с рисунком 3.15.

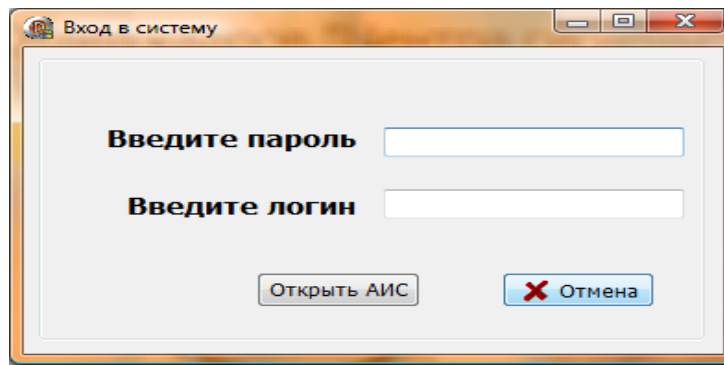


Рисунок 3.15 –Окно для ввода пароля и логина

Если на форме ввести неправильный логин и пароль, то появится сообщение, представленное в соответствии с рисунком 3.16.

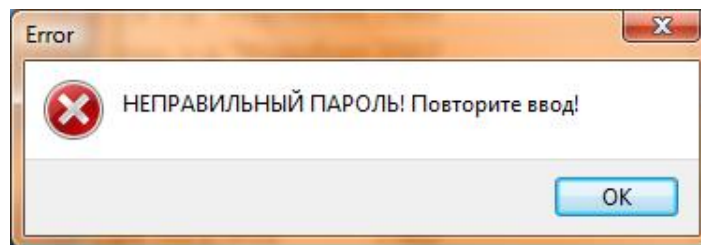


Рисунок 3.16 –Сообщение о неправильности ввода логина и пароля

При правильном вводе пароля откроется форма, где будут показаны данные о сотрудниках организации, представленная в соответствии с рисунком 3.17.

Код сотрудника	Фамилия	Имя	Отчество	Дата рождения	Специальность	Семейное положение	Возраст	Пол	Нация
1	Темботова	Елена	Николаевна	11.06.1969	врач-лаборант	замужем	43	женский	русская
2	Демченко	Виктория	Витиславовна	07.07.1975	врач-лаборант	замужем	37	женский	русская
3	Холина	Надежда	Дмитриевна	16.05.1956	помощник санитарног	замужем	56	женский	русская
4	Сушкова	Елена	Александровна	13.01.1956	помощник санитарног	не замужем	56	женский	русская
5	Праниченко	Ирина	Ивановна	15.01.1975	помощник санитарног	замужем	37	женский	русская
6	Сушкова	Елена	Николаевна	08.05.1960	врач-эпидимиолог	замужем	51	женский	русская
7	Щербак	Елена	Анатольевна	29.12.1971	фельдшер-лаборант	замужем	40	женский	русская
8	Воробьева	Вера	Анатольевна	03.08.1957	врач-бактериолог	не замужем	46	женский	русская
9	Кирсанова	Людмила	Николаевна	11.04.1969	фельдшер-лаборант	замужем	41	женский	русская
10	Рудавина	Надежда	Павловна	20.02.1967	лаборант-бактериоло	замужем	45	женский	русская
11	Кудрявцева	Валентина	Ивановна	21.09.1961	врач-лаборант	замужем	50	женский	русская
12	Пупынина	Зинаида	Александровна	23.02.1952	бухгалтер	замужем	59	женский	русская
13	Коваль	Ольга	Борисовна	21.04.1964	химик-эксперт	замужем	47	женский	русская
14	Венецова	Лилия	Григорьевна	07.03.1951	врач-лаборант	замужем	61	женский	русская
15	Кочетова	Надежда	Дмитриевна	15.09.1957	помощник санитарног	замужем	55	женский	русская
16	Юрченко	Ольга	Николаевна	14.03.1971	коммерсант	замужем	31	женский	русская

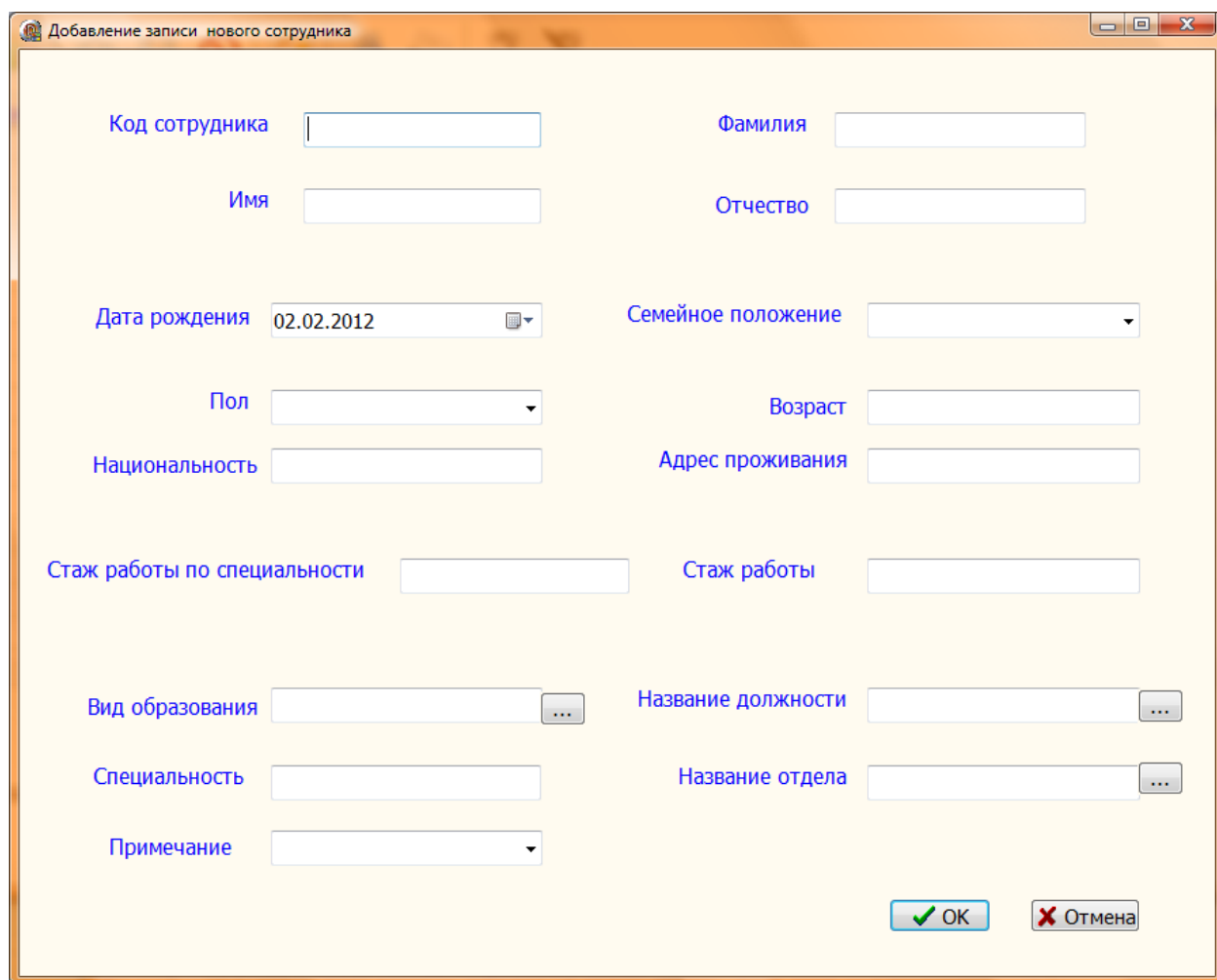
Поиск по

Введите код сотрудника

Рисунок 3.17 –Форма данных о сотрудниках

С помощью панели инструментов пользователь имеет возможность выполнить ряд функций.

Добавление записей в таблицу «Сотрудники». Для этого необходимо активировать значок на панели инструментов «Добавить сотрудника». В результате появится форма, представленная в соответствии с рисунком 3.18.В данную форму вводятся данные о новом сотруднике.



The screenshot shows a window titled "Добавление записи нового сотрудника" (Adding a new employee record). The form contains the following fields:

- Код сотрудника (Employee ID): text input field.
- Имя (Name): text input field.
- Фамилия (Surname): text input field.
- Отчество (Patronymic): text input field.
- Дата рождения (Date of birth): date picker showing 02.02.2012.
- Семейное положение (Marital status): dropdown menu.
- Пол (Gender): dropdown menu.
- Возраст (Age): text input field.
- Национальность (Nationality): text input field.
- Адрес проживания (Residence address): text input field.
- Стаж работы по специальности (Specialty work experience): text input field.
- Стаж работы (Work experience): text input field.
- Вид образования (Type of education): text input field with a browse button (...).
- Название должности (Job title): text input field with a browse button (...).
- Специальность (Specialty): text input field.
- Название отдела (Department name): text input field with a browse button (...).
- Примечание (Remarks): dropdown menu.

At the bottom right, there are two buttons: "OK" (with a green checkmark) and "Отмена" (Cancel, with a red X).

Рисунок 3.18 –Форма добавления записи нового сотрудника

Редактирование имеющихся данных в таблице «Сотрудники». Для этого необходимо активировать значок на панели инструментов «Редактировать сотрудника». В появившемся окне ввести нужные изменения информации и нажать кнопку «ОК». Результат выполненных действий представлен в соответствии с рисунком 3.19.

Рисунок 3.19 –Форма редактирования данных о выбранном сотруднике

Появится диалоговое окно, представленное в соответствии с рисунком 3.20. Для полного подтверждения действий в появившемся диалоговом окне следует нажать кнопку «Да».

Рисунок 3.20 –Диалоговое окно подтверждения редактирования данных

Удаление имеющихся данных в таблице «Сотрудники». Для этого необходимо активировать значок на панели инструментов «Удалить сотрудника». В появившемся диалоговом окне нажать кнопку «Да». Результат данной операции представлен в соответствии с рисунком 3.21.

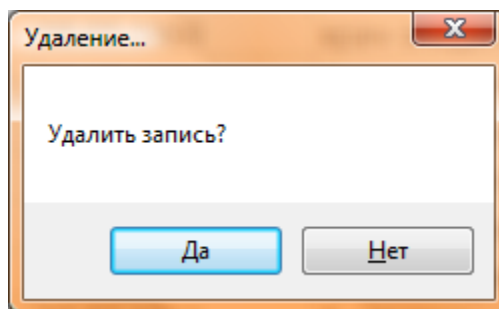


Рисунок 3.21 –Диалоговое окно подтверждения удаления данных

Поиск информации в таблице «Сотрудники». Для этого необходимо в поле ввода ввести фамилию необходимого сотрудника. В таблице останутся данные, удовлетворяющие условию поиска, представленные в соответствии с рисунком 3.22.

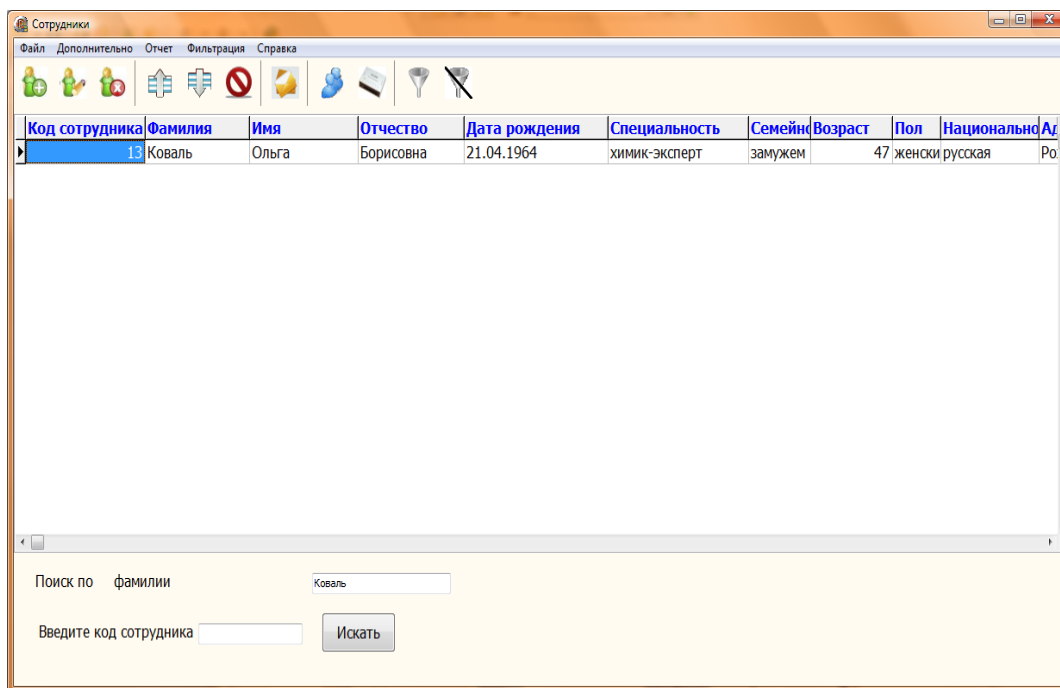


Рисунок 3.22 –Отображение записи, соответствующей условию поиска

Сортировка информации в таблице «Сотрудники» по возрастанию. Для этого необходимо выбрать столбец, который пользователь желает отсортировать и активировать значок на панели инструментов «Сортировка по возрастанию». Результат данной операции представлен в соответствии с рисунком 3.23.

Код сотрудника	Фамилия	Имя	Отчество	Дата рождения	Специальность	Семейн	Возраст	Пол	Национальн	Аг
17	Агафонова	Надежда	Викторовна	19.05.1955	бухгалтер	замужем	46	женски	русская	До
14	Веневцева	Лилия	Григорьевна	07.03.1951	врач-лаборант	замужем	61	женски	русская	Ан
8	Воробьева	Вера	Анатольевна	03.08.1957	врач-бактериолог	не замуж	46	женски	русская	Зе
2	Демченко	Виктория	Витиславовна	07.07.1975	врач-лаборант	замужем	37	женски	русская	Ли
20	Инкин	Виктор	Алексеевич	18.12.1975	экономист	женат	37	мужско	русский	Мс
9	Кирсанова	Людмила	Николаевна	11.04.1969	фельдшер-лаборант	замужем	41	женски	русская	Гр
13	Коваль	Ольга	Борисовна	21.04.1964	химик-эксперт	замужем	47	женски	русская	Ро
15	Кочетова	Надежда	Дмитриевна	15.09.1957	помощник санитарног	замужем	55	женски	русская	Ле
11	Кудряцева	Валентина	Ивановна	21.09.1961	врач-лаборант	замужем	50	женски	русская	за
5	Праниченко	Ирина	Ивановна	15.01.1975	помощник санитарног	замужем	37	женски	русская	Ми
12	Пульвина	Зинаида	Александровна	23.02.1952	бухгалтер	замужем	59	женски	русская	Сп
10	Рудавина	Надежда	Павловна	20.02.1967	лаборант-бактериоло	замужем	45	женски	русская	Ль
19	Сапрыкин	Владимир	Георгиевич	15.04.1950	преподаватель по во	женат	62	мужско	русский	Ли
4	Сущкова	Елена	Александровна	13.01.1956	помощник санитарног	не замуж	56	женски	русская	Су
6	Сущкова	Елена	Николаевна	08.05.1960	врач-эпидимиолог	замужем	51	женски	русская	Ли
1	Темботова	Елена	Николаевна	11.06.1969	врач-лаборант	замужем	43	женски	русская	На

Рисунок 3.23 –Результат сортировки выбранных данных

Аналогично выполняется сортировка по убыванию.

Фильтрация по дате рождения информации в таблице «Сотрудники». Для этого необходимо активировать значок на панели инструментов «Фильтрация по дате рождения». Появится диалоговое окно, представленное в соответствии с рисунком 3.24. В нем нужно указать диапазон фильтрации, после чего нажать кнопку «Ок».

Рисунок 3.24 –Диалоговое окно фильтрации по дате рождения

В результате в таблице останутся записи, удовлетворяющие диапазону фильтрации, представленные в соответствии с рисунком 3.25.

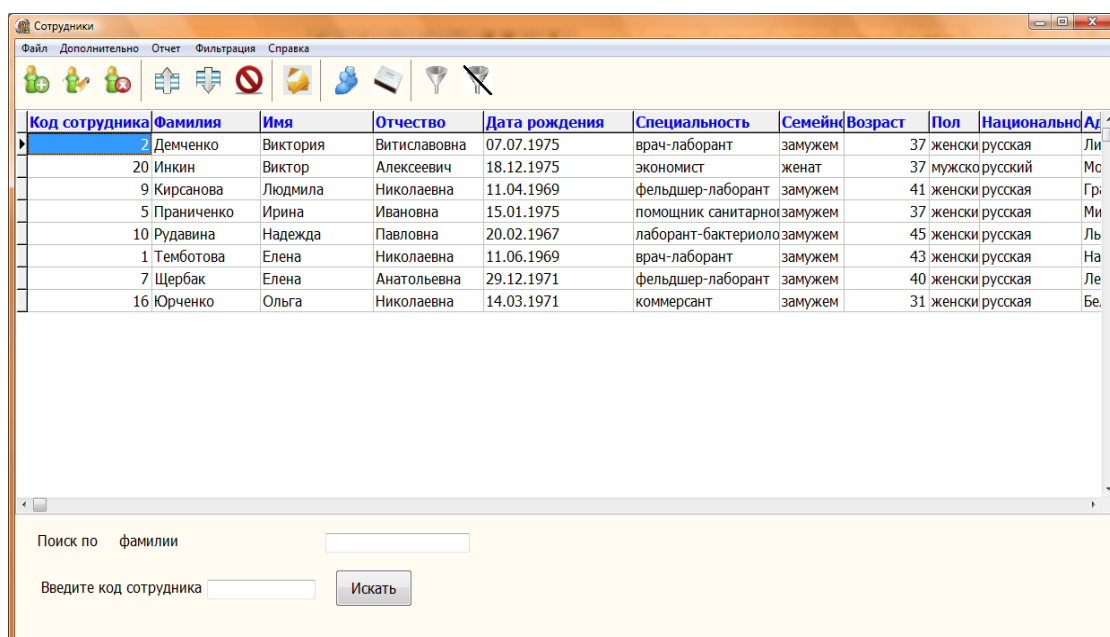


Рисунок 3.25 –Результат выполнения фильтрации по заданному диапазону

Просмотр информации о родственниках сотрудника в таблице «Сотрудники». Для этого необходимо активировать значок на панели инструментов «Сведения о родственниках». После чего появится окно с информацией о родственниках, представленное в соответствии с рисунком 3.26.

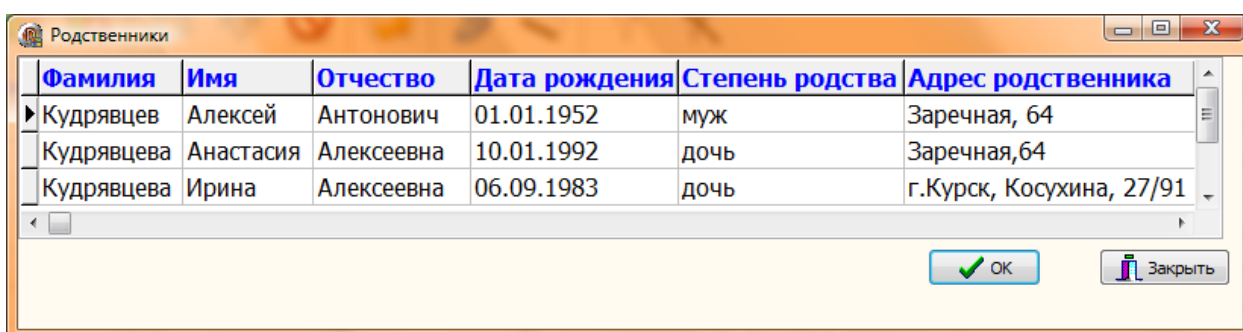
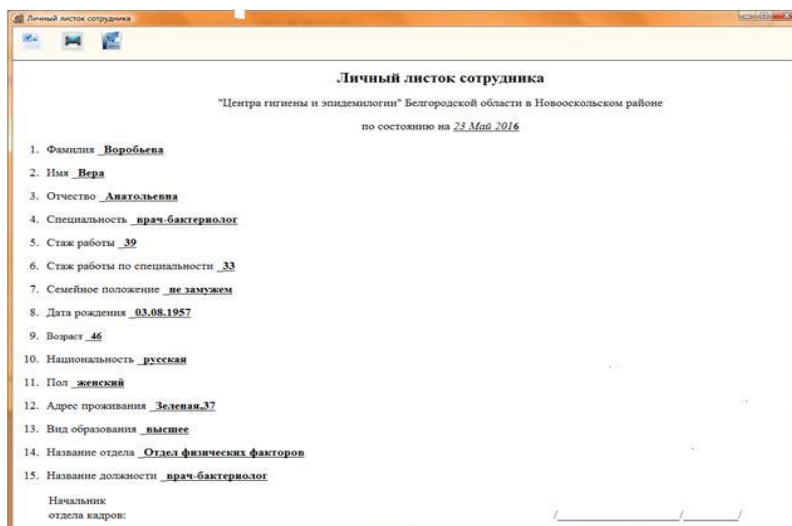


Рисунок 3.26 –Информация о родственниках сотрудника

Аналогично выполняется просмотр оборудования в таблице «Сотрудники». Для этого необходимо активировать значок на панели инструментов: «Сведения об оборудовании».

Вывод на печать «Личного листка сотрудника» в таблице «Сотрудники». Для этого необходимо активировать значок на панели инструментов «Личный

листок сотрудника». После чего появится окно с отчетом, представленное в соответствии с рисунком 3.27.



Личный листок сотрудника

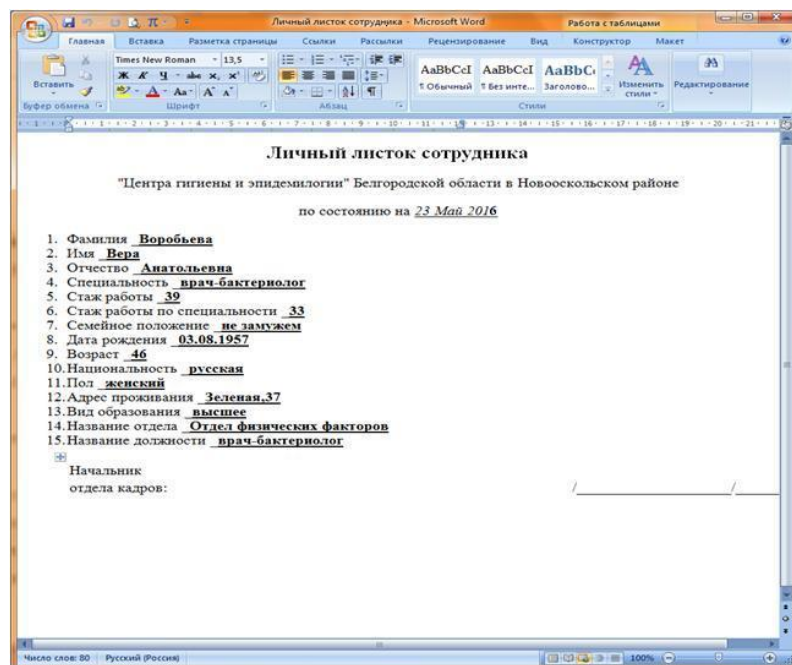
"Центра гигиены и эпидемиологии" Белгородской области в Новооскольском районе
по состоянию на 23 Май 2016

1. Фамилия Воробьева
2. Имя Вера
3. Отчество Анатольевна
4. Специальность врач-бактериолог
5. Стаж работы 39
6. Стаж работы по специальности 33
7. Семейное положение не замужем
8. Дата рождения 03.08.1957
9. Возраст 46
10. Национальность русская
11. Пол женский
12. Адрес проживания Зеленая,37
13. Вид образования высшее
14. Название отдела Отдел физических факторов
15. Название должности врач-бактериолог

Начальник
отдела кадров: _____

Рисунок 3.27 –Личный листок сотрудника

Для просмотра данного отчета в MicrosoftWord нужно нажать кнопку«Импорт в Word». В результате откроется окно MicrosoftWord, представленное в соответствии с рисунком 3.27.



Личный листок сотрудника

"Центра гигиены и эпидемиологии" Белгородской области в Новооскольском районе
по состоянию на 23 Май 2016

1. Фамилия Воробьева
2. Имя Вера
3. Отчество Анатольевна
4. Специальность врач-бактериолог
5. Стаж работы 39
6. Стаж работы по специальности 33
7. Семейное положение не замужем
8. Дата рождения 03.08.1957
9. Возраст 46
10. Национальность русская
11. Пол женский
12. Адрес проживания Зеленая,37
13. Вид образования высшее
14. Название отдела Отдел физических факторов
15. Название должности врач-бактериолог

Начальник
отдела кадров: _____

Рисунок 3.27–Импорт отчета в MSWord

Для настройки параметров печати нужно нажать кнопку –«Параметры печати». Для немедленной печати нужно нажать кнопку –«Печать».

Сообщения оператору выдаются в ходе выполнения функций редактирования и удаления данных в соответствии с рисунками

Выводы:

Разработано информационное обеспечение задачи, БД, программное обеспечение системы и ее технологическое обеспечение. Автоматизированная информационная система «Отдел кадров Центра гигиены и эпидемиологии Белгородской области Новооскольского района», предназначенная для обработки персональных данных о сотрудниках организации, их должностях и образовании была успешно протестирована.

4 Организационно-экономическая часть

4.1 Целесообразность разработки с экономической точки зрения

Эффективность управления человеческими ресурсами — это отношение результата (эффекта) к затратам.

Результатом управления человеческими ресурсами могут быть как факторы, достижение которых не связано с использованием компьютерных систем (далее КС), так и факторы, на которые КС оказывают существенное влияние. Мною затрагиваются только результаты, достигаемые с использованием КС, так как это тема ВКР.

Экономический эффект использования компьютеров в работе с человеческими ресурсами характеризуется количественным и качественным содержанием ее элементов. Качественные изменения в управлении влияют на количественные и непосредственно обуславливают их. При этом качественные изменения в процессе управленческого труда характеризуются снижением трудоемкости и ускорением сроков работ по обработке информации, повышением точности и достоверности, а также улучшением использования ее в практической работе с человеческими ресурсами.

Расчет экономической эффективности функционирования компьютерных систем предполагает сравнение полученного эффекта с затратами на его достижение. Основными факторами, обеспечивающими эффективность компьютеризации кадровых задач в сфере управления человеческими ресурсами, являются: высокая скорость выполнения операций по вводу, обработке и выводу кадровой информации; повышение качества учета, подбора и расстановки персонала за счет применения более современных методов и средств; сокращение сроков общей продолжительности решения кадровых задач и др.

При функционировании компьютерной системы указанные факторы влияют главным образом на снижение трудоемкости работ, повышение

производительности труда работников кадровых служб и создание творческой атмосферы в управлении предприятием. В настоящее время достаточно интенсивно развивается рынок современных технологий управления, в том числе человеческими ресурсами, которые активно применяются на предприятиях различных организационных структур, форм собственности и с разной численностью работников.

Накопленный отечественный и зарубежный опыт показывает, что компьютерные кадровые системы требуют значительных единовременных затрат, а также больших эксплуатационных расходов и определенных затрат живого труда работников. При этом важное значение придается установлению факторов, обеспечивающих эффективность систем, направлению воздействий указанных факторов, а также определению показателей их количественного измерения. К системе основных факторов, изложенных выше, следует добавить факторы повышения качества проведения вычислительных процедур в целом, а также повышения надежности функционирования вычислительных ресурсов, сокращения сроков обработки кадровой информации и др.

При определении эффективности компьютерных кадровых систем выявляется действие вышеупомянутых факторов на: результаты работы сотрудников кадровых служб и других подразделений предприятия; управленческую деятельность в целом по предприятию; процессы принятия кадровых решений при их информационно-вычислительной поддержке и др.

Для оценки эффективности компьютерных кадровых систем в управлении человеческими ресурсами требуется разработка соответствующей методики, способной определить отдачу компьютерной системы. При этом важно, чтобы методика позволила определить количественные величины эффективности производственного функционирования компьютерных кадровых систем в работе с человеческими ресурсами.

Под эффективностью компьютерной обработки кадровой информации понимают целесообразность применения средств вычислительной и организационной техники при вводе, хранении, корректировке и обработке

персональных сведений о человеческих ресурсах. При этом различают расчетную и фактическую эффективность компьютерных систем. Первая (расчетная) определяется на стадии создания КС, а вторая (фактическая) — по результатам ее внедрения в производственную эксплуатацию. Обобщенным критерием экономической эффективности КС являются минимальные затраты живого и овеществленного труда работников, занятых в сфере управления человеческими ресурсами предприятия.

Кроме того, экономический эффект КС ученые и специалисты, занятые их разработкой и внедрением, подразделяют на прямой и косвенный. Под прямой экономической эффективностью КС, как правило, понимают экономию материальных, трудовых ресурсов и финансовых средств, полученную в результате сокращения численности работников служб управления человеческими ресурсами. При этом прямой эффект выражается также в уменьшении фонда заработной платы работников, расхода материалов в результате компьютеризации конкретных кадровых задач и проведения анализа их решения. Отечественный и зарубежный опыт показывает, что на первой стадии работы КС количество работников, как правило, не сокращается. В этом случае учитывается косвенная эффективность КС, проявляющаяся в конечных результатах работы предприятия (например, в прибыли, улучшении психологического климата, сокращении текучести персонала и т.д.).

Критериями косвенной эффективности КС являются: сокращение сроков подготовки статистических отчетов, различных справочников и др.; повышение качества работы с человеческими ресурсами, например отсутствие конфликтов, рекламаций на продукцию и др.; повышение управленческой культуры работников кадровых служб предприятия.

Основным же показателем является значительное повышение качества управления человеческими ресурсами, которое, как и при прямой экономической эффективности, приводит к значительной экономии живого и овеществленного труда.

Определяют экономическую эффективность КС с помощью трудовых и

стоимостных показателей. Данный подход разработан М.И.Семеновым, И.Т.Трубилиным и др. Он основан на методе сопоставления данных базисного и отчетного периодов. За базисный период принимают затраты на обработку кадровой информации до внедрения КС (при обработке вручную), а при внедрении КС — затраты на обработку этой информации.

Так, например, на ручную обработку кадровой информации по составлению различных аналитических отчетов следует затратить 100 чел./ч (Т₀), а при использовании КС — 5 чел./ч (Т₁).

Относительный индекс производительности труда работников кадровых служб (И_{пт}) будет:

$$И_{пт} = 0,05 . \quad (4.1)$$

Это означает, что для подготовки таких аналитических отчетов при компьютеризации требуется по сравнению с ручной обработкой всего 5% времени.

Используя И_{пт}, определяют относительный показатель экономии трудовых затрат в службе управления человеческими ресурсами предприятия. А при обработке этой информации на компьютерах экономия составляет 95%. Однако наряду с трудовыми показателями необходимо определить и стоимостные показатели, т. е. определить все затраты на обработку кадровой информации в базисном и отчетном вариантах в денежных единицах.

Методика, позволяющая рассчитать экономическую эффективность использования и окупаемость КС в управлении человеческими ресурсами предприятия, более сложная. Она была разработана И.А. Пиюковым и находит применение на ряде предприятий России, где создаются, внедряются и используются компьютерные кадровые системы.

Методика оценки экономической эффективности обработки информации в компьютерных системах. В предлагаемой методике сравниваются значения показателей ручной и компьютерной обработки. Годовая экономия (прибыль) (Э_г) определяется как разность между затратами на обработку информации вручную (до внедрения КС) (З_р) и затратами на

компьютерную обработку информации (Z_k).

$$\Delta_r = Z_p - Z_k. \quad (4.2)$$

Затраты на обработку информации до внедрения современных технологий складываются из затрат на все ручные операции:

$$Z_p = T_p V_p (1 + Д + Н), \quad (4.3)$$

где T_p — трудоемкость ведения ручной картотеки (массива) и составления различных отчетов, ч/год;

V_p — средняя часовая ставка сотрудника кадровой службы, руб.;

Д — дополнительная заработная плата и различные отчисления;

Н — накладные расходы.

Трудоемкость ведения ручной картотеки и составления отчетов определяется на основе существующих норм времени на оформление документов по персоналу на производственных предприятиях:

$$T_p = \sum_{i=1}^n \frac{Q \cdot h_i \cdot t}{10}, \quad (4.4)$$

где Q — общее количество (объем) личных карточек, шт.;

t — норма времени на обработку 10 карточек при составлении одного отчета (50 мин);

h_i — периодичность выдачи в год i -й формы;

i — вид отчета ($i = 1, 2, 3, \dots, n$);

n — количество отчетов.

При использовании учетного документа, отличного от утвержденной формы, норма времени (t_1), для обработки 10 форм определяется в минутах по формуле:

$$t_1 = \frac{P}{P_q}, \quad (4.5)$$

где P — количество кадровых показателей учетного документа,

отличного от стандартной формы, ед.;

P_q — количество кадровых показателей учетного документа формы, ед.

Затраты необходимых средств на обработку кадровой информации после внедрения современных технологий определяются на основе затрат времени на выполнение ручных, машинно-ручных и компьютерных операций. К ручным операциям относятся: заполнение первичных (входных) документов, преобразование кадровой информации в требуемую форму, логический контроль информации, регистрация приема и выдачи. К машинно-ручным операциям относится подготовка промежуточных компьютерных носителей информации. Затраты времени на машинно-ручные операции (T_m) определяются по формуле:

$$T_m = \frac{Q_{вв}}{N}, \quad (4.6)$$

где $Q_{вв}$ — объем вводимой информации, символов;

N — норма выработки при подготовке компьютерного носителя, символов/ч.

К компьютерным операциям относятся те, которые выполняются непосредственно компьютером и соответствующим программным продуктом. Количество компьютерного времени ($T_{ком}$) необходимое для проведения одного сеанса, определяется по формуле:

$$T_{ком} = T_{совм} + T_{вв} + T_{обр} + T_{выв}, \quad (4.7)$$

где $T_{совм}$ — время совмещения работы всех устройств компьютера;

$T_{вв}$ — время ввода информации в компьютер;

$T_{обр}$ — время обработки кадровой информации (решение задачи);

$T_{выв}$ — время, затрачиваемое на вывод результатов обработки (решение задачи).

Количество времени, необходимое для совмещения работы устройств

компьютера ($T_{совм}$), определяется по формуле:

$$T_{совм} = \sum_{i=1}^K R_i, \quad (4.8)$$

где K — количество устройств компьютера;

R_i — время совмещения (подготовки) одного устройства.

Время ввода кадровой информации ($T_{вв}$) определяется по формуле:

$$T_{вв} = \sum_{i=1}^m \frac{Q_{вви}}{S_{вви}}, \quad (4.9)$$

где $Q_{вви}$ — объем кадровой информации, вводимой в компьютер;

$S_{вви}$ — быстродействие компьютера;

m — количество устройства ввода.

Время обработки информации определяется по формуле:

$$T_{ком} = O_{ay} \cdot T_{ay} \cdot O_{оби} \cdot T_{оби}, \quad (4.10)$$

где O_{ay} — количество операций, выполняемых компьютером;

T_{ay} — время выполнения операции;

$O_{оби}$ — количество обращений к компьютеру;

$T_{оби}$ — среднее время обращения к компьютеру.

Время, необходимое для выводов результатов на печать ($T_{выв}$), определяется по формуле:

$$T_{выв} = \sum_{i=1}^t \frac{Q_{выви}}{S_{выви}}, \quad (4.11)$$

где $Q_{выви}$ — объем информации, выводимой компьютером;

$S_{выви}$ — быстродействие компьютера;

t — количество устройств вывода.

Окончательная стоимость обработки кадровой информации (затраты средств) (Z_m) определяется по формуле:

$$Z_M = T_p \cdot t_{чр} + T_M \cdot t_{чм} + T_{ком} \cdot C_{мч}, \quad (4.12)$$

где $t_{чр}$ — средняя часовая тарифная ставка работника при выполнении ручных операций;

$t_{чм}$ — средняя часовая тарифная ставка работника при выполнении машинно-ручных операций;

$C_{мч}$ — стоимость одного часа работы ЭВМ, руб.

Единовременные затраты на применение компьютерных технологий в работе с кадровыми (K_a) рассчитываются по формуле

$$K_a = K_p + K_{ком}, \quad (4.13)$$

где K_p — затраты на создание, руб.;

$K_{ком}$ — доля стоимости компьютера, приходящаяся на решение данной кадровой задачи, руб.

В свою очередь

$$K_{ком} = \frac{\text{Стоимость_компьютера}}{T_{полез}}, \quad (4.14)$$

где $T_{полез}$ - годовой фонд полезного компьютерного времени.

Получив общую сумму единовременных затрат и снижение стоимости обработки кадровой информации, рассчитывают годовой экономический эффект ($\mathcal{E}_{эф}$) по следующей формуле:

$$\mathcal{E}_{эф} = \mathcal{E}_r - E_n \cdot K_p, \quad (4.15)$$

где E_n — нормативный коэффициент экономической эффективности капитальных вложений.

Срок окупаемости затрат (T_ϕ) определяется по формуле

$$T_\phi = \frac{K_p}{K_{ком}}. \quad (4.16)$$

Расчетный коэффициент (E_p) экономической эффективности

определяется по формуле:

$$E_p = \frac{1}{T_\phi}. \quad (4.17)$$

Нормативный срок окупаемости (T_n) по компьютерным технологиям в управлении человеческими ресурсами принят равным не более 3,3 года. Компьютерная технология будет эффективной, если:

$$\begin{aligned} T_\phi &\leq T_n, \\ E_p &\geq E_{\text{нвт}}, \end{aligned}$$

где $E_{\text{нвт}}$ — нормативный коэффициент экономической эффективности затрат на компьютерную технику для всех отраслей равен 0,321 (получен на основе экспертных оценок).

4.2 Калькуляция себестоимости научно-технической продукции

4.2.1 Расчёт трудозатрат на разработку программного продукта

Произведём расчёт трудозатрат.

Трудозатраты измеряются в чел. • час.

$$T = t_{\text{и}} + t_{\text{а}} + t_{\text{п}} + t_{\text{отл.окон.}} + t_{\text{д}}, \quad (4.18)$$

где: $t_{\text{и}}$ – затраты труда на исследование алгоритма решения задачи;

$t_{\text{а}}$ – затраты на разработку блок-схемы алгоритма;

$t_{\text{п}}$ – затраты на программирование;

totл. окон.– затраты на отладку программы;

$t_{\text{д}}$ – затраты на подготовку документации.

Затраты труда на исследование алгоритма решения задачи с учетом уточнения описания и квалификации программиста вычисляются по формуле:

$$t_{\text{и}} = Q \cdot B / ((75..85) \cdot k), \quad (4.19)$$

где: Q – количество отчетов в программе;

V – коэффициент увеличения затрат зависит от сложности программы (1...5);

k – коэффициент квалификации разработчика.

$$Q = q \cdot c \cdot (1 + p) \cdot v. \quad (4.20)$$

$$Q = 10 \cdot 1 \cdot (1+1) \cdot 100 = 1100 \text{ шт.}$$

где: q – предполагаемое число отчетов и форм;

c – коэффициент сложности программы (1...2);

p – коэффициент коррекции программы в ходе ее разработки (0,5..1);

v – нормировочный коэффициент (100).

Коэффициент. увеличения затрат V характеризует увеличение затрат труда вследствие недостаточно полного описания задачи, уточнений и некоторой доработки. Этот коэффициент может принимать значения от 1,2 до 5.

Коэффициент квалификации разработчика зависит от стажа работы программиста следующим образом:

- стаж до 2-х лет – k=0,8;

- от 2 до 3 лет – k=1;

- от 3 до 7 лет – k=1,3...1,4;

- от 7 лет – k=1,5...1,6.

Произведём расчёт затраты труда на исследование алгоритма решения задачи:

$$t_{\text{и}} = \frac{(1100 \cdot 3)}{85 \cdot 0,8} = 29,64 \text{ чел. час.}$$

Затраты труда на исследование алгоритма решения задачи составят 47,07чел.*час.

Расчет затрат на разработку блок-схемы алгоритма.

Расчет трудозатрат производится по формуле:

$$t_a = Q / (60 \dots 75) \cdot k, \quad (4.21)$$

$$t_a = 1100/(75 \cdot 0,8) = 11,73 \text{ чел.} \cdot \text{ час.}$$

Затраты на разработку блок-схемы алгоритма составят 11,73 чел.*час. Расчет затрат на отладку программы на персональном компьютере. Отладкой программ занимается программист.

Отладка программы – выполнение программы для выявления дефектов в функциях, в логике, проводится проверка программного продукта на соответствие технического задания.

Расчет трудозатрат на отладку производится по формуле:

$$T_{\text{отл}} = Q/(40 \dots 50) \cdot k, \quad (4.22)$$

$$T_{\text{отл}} = 1100/(40 \cdot 0,8) = 34,38 \text{ чел.} \cdot \text{ час.}$$

Затраты на отладку программы на персональном компьютере составят 34,38 чел.*час.

При комплексной отладке программы следует предусмотреть возрастающие в 1,5 раз затраты, поэтому окончательные трудовые затраты на отладку программы будут равны:

$$t_{\text{отл.окон.}} = t_{\text{отл}} \cdot 1,5, \quad (4.23)$$

$$t_{\text{отл.окон.}} = 34,38 \cdot 1,5 = 51,56 \text{ чел.} \cdot \text{ час.}$$

Затраты на окончательную отладку программы на персональном компьютере составят 51,56 чел. *час.

Трудовые затраты на подготовку документации будут складываться из затрат труда на подготовку рукописного текста и затрат труда на редактирование, печать и оформление документации.

$$t_d = t_{\text{дп}} + t_{\text{др}}, \quad (4.24)$$

где: $t_{\text{дп}}$ – трудовые затраты на подготовку материалов в рукописи;

$t_{\text{др}}$ – затраты на редактирование, печать и оформление документов.

Рассчитаем трудозатраты на подготовку материалов в рукописи по формуле:

$$t_{дп} = \frac{Q}{(150...200)*k}, \quad (4.25)$$

$$t_{дп} = 1100/(200*0,8) = 6,87 \text{ чел.} \cdot \text{ час.}$$

Затраты на редактирование, печать и оформление тдр прямо пропорционально зависят от затрат на подготовку материалов в рукописи.

$$t_{др} = 0,75 \cdot t_{дп}, \quad (4.26)$$

$$t_{др} = 0,75 \cdot 6,87 = 5,15 \text{ чел.} \cdot \text{ час.}$$

Произведём расчёт трудовых затрат на подготовку документации:

$$t_{д} = 87 + 5,15 = 12,02 \text{ чел.} \cdot \text{ час.}$$

Таким образом, общие трудовые затраты на подготовку документации составят 12,02 чел. час.

Расчет трудозатрат на написание исходного кода программы (программирование).

Программирование – процесс создания компьютерных программ и/или программного обеспечения с помощью языков программирования.

Затраты на программирование составляют примерно 20-30% от общих трудозатрат на остальные этапы разработки программы.

Рассчитаем затраты на программирование, как 20% от общих трудозатрат:

$$t_{п} = (t_{и} + t_{а} + t_{отл.окон.} + t_{д}) \cdot 0,20, \quad (4.27)$$

$$t_{п} = (29,64 + 11,73 + 51,56 + 12,02) \cdot 0,20 = 20,99 \text{ чел.} \cdot \text{ час.}$$

Затраты на программирование составят 20,99 чел. час.

Расчет общих трудозатрат на разработку программного продукта.

Полученные трудозатраты сводим в таблицу, производим суммирование всех видов трудозатрат на разработку программного и находим общие трудозатраты Т:

$$T = 29,64 + 11,73 + 51.56 + 12.02 + 20.99 = 125.94 \text{ чел.} \cdot \text{ час.}$$

Таким образом, общие трудозатраты на разработку программного продукта составят 125.94 чел.•час.

Из приведенной таблицы 4.1 видно, что наибольшее количество трудозатрат при разработке программного продукта обычно вызывают: на исследование алгоритма и отладку программы.

Таблица 4.1 – Структура трудозатрат на разработку программного продукта

Виды трудозатрат	Индекс	Трудоемкость, чел.•час	Структура трудозатрат, %
На исследование алгоритма	$t_{и}$	29,64	23.53
На разработку блок – схемы	$t_{а}$	11,75	9.32
На программирование	$t_{п}$	20.99	16.66
На отладку программы	$T_{отл.окон}$	51.56	40.94
На подготовку документации	$t_{д}$	12.02	9.55
Итого (общие трудозатраты)	T	125.94	100

4.2.2 Составление сметы затрат на разработку программного продукта

Смета затрат на разработку программного продукта включает в себя следующие статьи:

- затраты на оплату труда программисту;
- единый социальный налог (26 % от затрат на оплату труда);
- затраты на оплату машинного времени.

Расходы на оплату труда разработчика программы ($ЗП_{пр}$) определяются путем умножения трудоемкости создания программы на среднюю часовую зарплату программиста. Запишем расходы на оплату труда разработчика программы в виде формулы:

$$ЗП_{\text{осн}} = Т \cdot СЧ_{\text{пр}} , \quad (4.28)$$

где: Т–трудоемкость разработки программного продукта, чел.•час;

СЧ_{пр}– средняя часовая оплата труда программиста, рублей в час.

$$СЧ_{\text{пр}} = \frac{П_{\text{пр}}}{\Phi_{\text{рв}}}, \quad (4.29)$$

где: П_{пр}– заработная плата программиста;

Φ_{рв}– месячный фонд рабочего времени, часов.

Произведём расчёт среднечасовой оплаты труда программиста:

$$СЧ_{\text{пр}} = 5600/169,2 = 33,09 \text{ руб./час.}$$

Произведём расчёты оплаты труда разработчика программы:

$$ЗП_{\text{осн}} = 125,94 \cdot 33,09 \approx 4167,35 \text{ руб.}$$

Дополнительная заработная плата:

$$ЗП_{\text{доп}} = ЗП_{\text{осн}} \cdot \frac{15}{100}, \quad (4.31)$$

$$ЗП_{\text{доп}} = 4167,35 \cdot 15/100 \approx 625,10 \text{ руб.}$$

Итого затраты на оплату труда:

$$ЗП_{\text{общ}} = ЗП_{\text{осн}} + ЗП_{\text{доп}} , \quad (4.30)$$

$$ЗП_{\text{общ}} = 4167,35 + 625,10 = 4792,45 \text{ руб.}$$

Расходы на оплату труда разработчика программы составят 4792.45 руб.

Произведем расчет отчислений в социальные фонды. Единый социальный налог составляет 26 % от затрат на оплату труда программиста:

$$ЕСН = ЗП_{\text{общ}} \cdot 0,26 \quad (15) \quad ЕСН = 4792,45 \cdot 0,26 \approx 1246,03 \text{ руб.}$$

Рассчитаем затраты на оплату машинного времени при отладке программы. Стоимость машинного часа работы рассчитывается по формуле:

$$С_{\frac{м}{ч}} = А_{мк} + С_{эл} , \quad (4.32)$$

где: С_{м/ч}– стоимость машинного часа работы, руб.;

$A_{мк}$ – амортизация компьютера за 1 м/ч, руб.;

$C_{эл}$ – стоимость электроэнергии за 1 ч работы, руб.

Таблица 4.2 – Рыночная стоимость компьютера

Наименование	Марка	Стоимость, руб.
Ноутбук	Acer series N53S	19400
Мышка	Мышь Genius Net Scroll	600
Итого:		20000 руб.

Амортизация компьютера за 1 м/ч вычисляется по следующей формуле:

$$A_{мк} = C_{к} \cdot N_{а} / (T \cdot 100\%), \quad (4.33)$$

где: $C_{к}$ – стоимость компьютера, руб.;

$N_{а}$ – норма амортизации компьютера.

$$A_{мк} = \frac{20000 \cdot 15\%}{125.94 \cdot 100\%} = \frac{300000}{12594} \approx 23.8 \text{ руб.}$$

Стоимость электроэнергии за 1 ч работы вычисляется по следующей формуле:

$$C_{эл} = M_{к} \cdot C_{кв}, \quad (4.34)$$

где: $M_{к}$ – мощность компьютера, кВт/ч;

$C_{кв}$ – стоимость 1 кВт электроэнергии, руб.

Рассчитаем стоимость электроэнергии.

$$C_{эл} = 0,4 \cdot 3,23 = 1,29 \text{ руб.}$$

Используя теперь полученные значения, можно рассчитать стоимость 1 м/ч:

$$C_{м/ч} = 23,8 + 1,29 = 25,09 \text{ руб.}$$

Себестоимость программы:

$$C_{пр} = C_{м/ч} \cdot T + 3П_{общ} + ЕСН, \quad (4.35)$$

$$C_{пр} = 25,09 \cdot 125.94 + 4792.45 + 1246.03 = 9198,31 \text{ руб.}$$

4.2.3 Обоснование цены программного продукта

Произведём расчёт цены программного продукта и её обоснование.

Примем процент рентабельности (условно) – 15%

Прибыль рассчитать по формуле:

$$П = \frac{C_{\text{полн}} \cdot \%P}{100}, \quad (4.36)$$

где: $C_{\text{полн}}$ – себестоимость;

P – процент рентабельности.

$$П = (9198,31 \cdot 15)/100 = 1379,74 \text{руб.}$$

Цена программного продукта равна сумме полной себестоимости и прибыли:

$$Ц = C_{\text{полн}} + П, \quad (4.37)$$

$$Ц = 9198,31 + 1379,74 = 10578,05 \text{руб.}$$

Цена программного продукта с НДС (НДС в России 18%):

$$\text{НДС} = \frac{Ц \cdot 18}{100},$$

$$\text{НДС} = (10578,05 \cdot 18)/100 = 1904,04 \text{руб.} \quad (4.38)$$

Рассчитаем стоимость продукта по отпускной цене:

$$Ц_{\text{отп}} = Ц + \text{НДС}, \quad (4.39)$$

$$Ц_{\text{отп}} = 10578,05 + 1904,04 = 12482,09 \text{руб.}$$

Стоимость программного продукта составляет 12482,09 руб.

4.2.4 Расчет экономической эффективности

Значение экономической эффективности от использования программного продукта определяется по формуле:

$$\mathcal{E} = Z_{\text{б}} - Z_{\text{п}}, \quad (4.40)$$

где: $Z_{\text{б}}$ – затраты по базовому варианту, руб.;

$Z_{\text{п}}$ – затраты при использовании программного обеспечения (ПО), руб.

$$Z_6 = CЧ_{\text{пр}} \cdot T_p \cdot \left(\frac{1}{d3\Pi}\right), \quad (4.41)$$

где: T_p – трудоемкость разработки программного продукта, чел*час;

$CЧ_{\text{пр}}$ – средняя часовая оплата труда программиста, рублей в час.

$d3\Pi$ – доля заработной платы в общей смете затрат организации, определяется:

$$d3\Pi = \frac{4792,45 \cdot 100}{9198,31} \approx 52,10\%.$$

Для решения задачи без использования разрабатываемого программного продукта необходимо примерно в 3 раза больше действующего фонда рабочего времени.

$$T_p = T \cdot 3, \quad (4.42)$$

$$T_p = 125,94 \cdot 3 = 377,82 \text{руб. час.}$$

Теперь определим затраты без использования программы:

$$Z_6 = 33,09 \cdot 377,82 \cdot \left(\frac{1}{52,10} \cdot 100\right) = 23996,28 \text{(руб)}$$

где: $C_{\text{м/ч}}$ – стоимость машинного часа работы, руб.;

T_c – срок службы программного обеспечения, принимаемым равным 5 годам.

Произведём расчёты затрат при использовании программы:

$$Z_9 = \frac{(377,82 \cdot 25 + 9198,31)}{5} = 3728,76 \text{руб.}$$

Теперь рассчитаем значение экономического эффекта от использования программного продукта:

$$\Delta = 23996,28 - 3728,76 = 20267,52 \text{руб.}$$

Экономическая эффективность системы составит около 160%.

Определим срок окупаемости капиталовложений:

$$T_{\text{ср}} = \frac{20000 + 9198,31}{20267,52} \approx 1,4 \text{ года.}$$

Таблица 4.3 – Основные технико-экономические показатели

Показатели	Ед. измерения	Значения
Общие трудозатраты на разработку программного продукта	Чел. • час	125,94
Затраты на оплату труда	Руб.	4792,45
Отчислений в социальные фонды	Руб.	1246,03
Затраты на оплату машинного времени	Руб.	23,08
Себестоимость программы	Руб.	9198,31
Цена программного продукта	Руб.	12482,09
Экономический эффект	Руб.	20267,52
Экономическая эффективность	%	160
Срок окупаемости капиталовложений	Год	1,44

Выводы:

Таким образом, произведенный экономический анализ эффективности создания и эксплуатации программного продукта доказывает целесообразность его использования на предприятии.

ЗАКЛЮЧЕНИЕ

В данной работе создана автоматизированная информационная система «Отдел кадров центра гигиены и эпидемиологии Белгородской области Новооскольского района», предназначенная для автоматизации обработки персональных данных о сотрудниках организации.

Автоматизированная информационная система создавалась по заявке филиала ФБУЗ «Центр гигиены и эпидемиологии» Новооскольского района Белгородской области.

В процессе разработки было решено множество задач, начиная от исследования предметной области до разработки руководств системного программиста и оператора.

Для проектирования и разработки автоматизированной информационной системы использовалась система быстрой разработки приложение RAD Studio и система управления базами данных Microsoft Access.

Разработанная автоматизированная информационная система «Отдел кадров центра гигиены и эпидемиологии Белгородской области Новооскольского района» автоматизирует обработку и управление информацией о сотрудниках организации, их должностях и образовании, об оборудовании, относящиеся к сотруднику.

Автоматизированная информационная система «Отдел кадров центра гигиены и эпидемиологии Белгородской области Новооскольского района» используется начальником отдела кадров филиала ФБУЗ «Центр гигиены и эпидемиологии» Новооскольского района Белгородской области.

В первой главе рассмотрена характеристика объекта исследования филиала ФБУЗ «Центр гигиены и эпидемиологии» Новооскольского района Белгородской области, определена экономическая сущность задачи, обоснована необходимость использования вычислительной техники для решения поставленных в ВКР задач. Кроме того, в рамках первой главы был проведен

сравнительный анализ существующих разработок в области автоматизации кадрового учета.

Во второй главе приведено обоснование и выбор проектных решений по техническому, информационному и программному обеспечению разрабатываемой системы. А также проведено обоснование выбора программных средств.

В третьей главе разработано информационное обеспечение задачи, БД, программное обеспечение системы и ее технологическое обеспечение. Кроме того, был описан контрольный пример реализации проекта.

В четвертой главе был произведен экономический анализ эффективности создания и эксплуатации программного продукта, который доказывает целесообразность его использования на предприятии

Цель ВКР достигнута. Для достижения данной цели были решены следующие задачи:

- выбрано программное обеспечение для разработки АИС;
- разработаны информационные модели с использованием программных и аппаратных средств;
- разработан программный продукт;
- тестирование и отладка разработанного программного продукта.

СПИСОК ИСПОЛЬЗОВАННЫХ ИСТОЧНИКОВ

- 1 MicrosoftOfficeAccess 2010. Шаг за шагом:[Электронные ресурсы]Пер. с англ. -М.: СП ЭКОМ, 2013. - 431 с.
- 2 Автоматизированные информационные системы, базы и банки данных. Вводный курс: Учебное пособиеМ.: [Электронные ресурсы] Гелиос АРВ, 2014. – 368 с.
- 3 Агальцов, В.П. Базы данных[Текст] /В.П. Агальцов. // М.: Мир, 2011. – 376 с.: ил. + CD с примерами.
- 4 Боровский, А Программирование в Delphi. [Текст] / А. Боровский // Издат. БХВ-Петербург, 2014. – 448 с.
- 5 Архангельский, А.Я. Программирование в Delphi [Текст] / А.Я.Архангельский // М.:ООО Издательство «БИНОМ», 2014. - 1152 с, ил.
- 6 Банк, В.Р. Информационные системы в экономике [Текст] /В.Р. БанкВ.С. Зверев // М.:Экономистъ, 2010. – 420 с.
- 7 Бархударов, А.С. Delphi 7: учебный курс. [Текст] / А.С. Бархударов //СПб. Издательство «Питер», 2012. - 543 с, ил.
- 8 Бекаревич, Ю.Б. В. MS Access 2010 за 30 занятий. [Текст] / Ю. Б.Бекаревич,Н. В. Пушкина // СПб: БХВ-Петербург, 2011. - 492 с.
- 9 Глушаков, С.В., Базы данных: Учебный курс.[Текст] / С.В.Глушаков, Д.В.Ломотько // Феникс; Киев: Абрис, 2010. – 504 с.
- 10 Голицина, О.Л. «Базы данных». [Текст] /О.Л.Голицина,Н.В.Максимов, И.И.Попов. // М.: Диалектика, 2013 г. – 350 с.
- 11 Гофман, В.Э., Delphi. Быстрый старт. [Текст] / В.Э.Гофман, А.Д. Хомоненко // СПб.: БХВ-Петербург, 2011. – 288 с., ил.
- 12 Гофман, В.Э. Работа с базами данных в Delphi.[Текст] /В.Э. Гофман, А.Д. Хомоненко // СПб.: БХВ-Петербург, – 2-е изд. –2009. – 624 с.: ил.
- 13 Дарахвелидзе, П. Г. Программирование в Delphi 7 [Текст] / П.Г. Дарахвелидзе,Е. П.Марков // СПб.: БХВ-Петербург, 2011. —784 с.: ил.

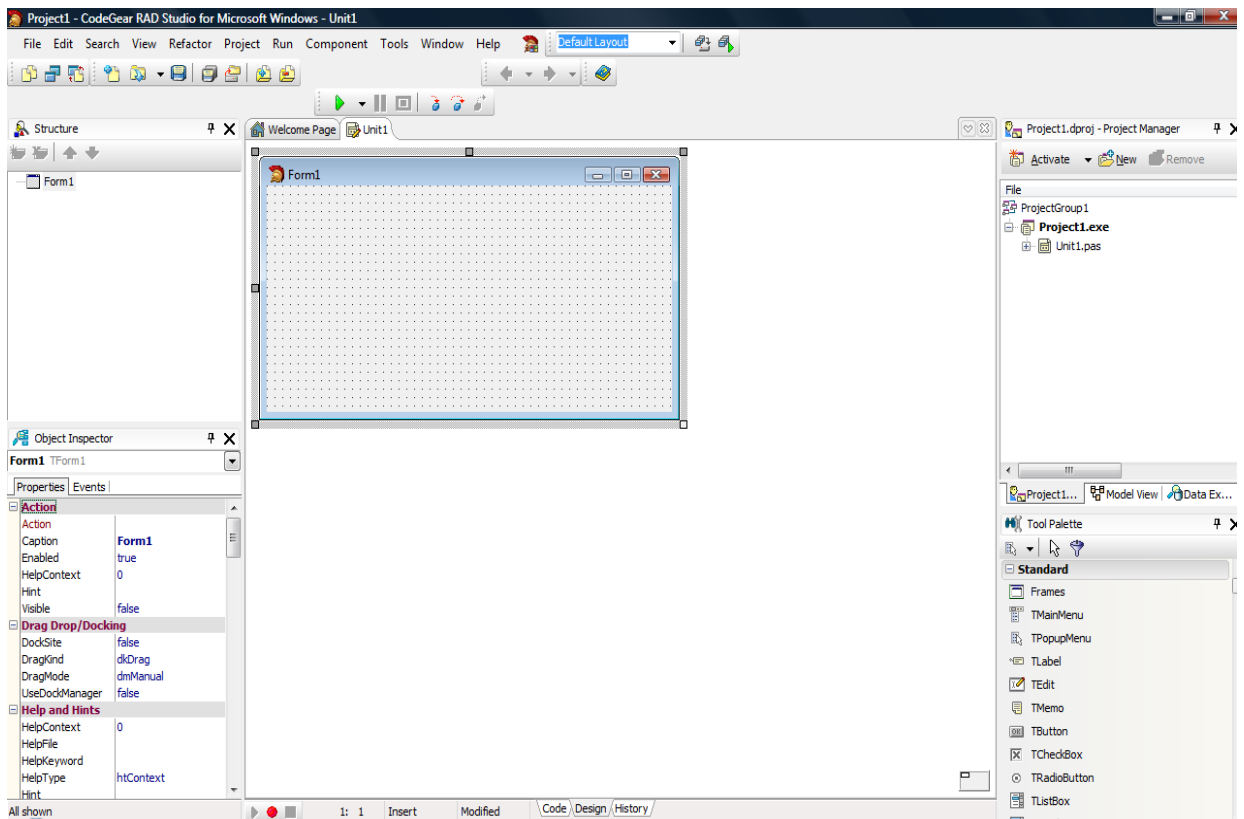
- 14 Дарахвелидзе, П.Г., Delphi 7 - среда визуального программирования.[Текст] /П.Г. Дарахвелидзе,Е.П. Марков // СПб.: ВHV - Санкт-Петербург, 2010. - 352 с.
- 15 Дубнов, П.Ю. Access 2010: Самоучитель. [Текст] / П.Ю. Дубнов// М.: ДМК Пресс, 2010.- 313 с.
- 16 Зубков, А. П. Программирование на Delphi 7.[Текст] / А.П.Зубков // СПб.: Издательство «Питер», 2011. – 396 с, ил.
- 17 Карпов, Б. Microsoft Access 2010: Справочник. [Текст] / Б.Карпов // СПб.:Питер, 2011. – 416с.
- 18 Ковалев, Е.В. Программирование в Delphi 7. [Текст] / Е.В.Ковалев // СПб.: БХВ-Петербург, 2012. – 590 с., ил.
- 19 Комолова, Н.В., HTML:Учебный курс.[Текст] / Н.В. Комолова// СПб.: Питер, 2013. – 268с.: ил.
- 20 Конноли, Т.К. «Базы данных проектирование реализация и сопровождение».[Текст] / Т.К. Комолова // Берг: Диалектика, 2010 г.-180 с.
- 21 Крэнке, Д. Теория и практика построения баз данных. [Текст] / Д. Крэнке // СПб.: Питер, 2010, 800 с.
- 22 Кузин А.В., Левонисова С.В. Базы данных: учеб.пособие для студ. высш. учеб. заведений[Текст] / А.В. Кузин, С.В., Левонисова // М.: Издательский центр «Академия», 2011. – 320с.
- 23 Культин Н.Б. Основы программирования в Delphi 7. [Текст]/ Н.Б. Культин // СПб.: БХВ-Петербург, 2012. – 608 с., ил.
- 24 Лаптин, А.А. Delphi 7: учебный курс.[Текст] / А.А. Лаптин // СПб.: Издательство «Питер», 2012. - 651 с, ил.
- 25 Липаев, В.В. Отладка сложных программ: Методы, средства, технология.[Текст] / В.В. Липаев // М.: ООО Издательство «АСТ», 2015. – 384 с.
- 26 Марков, Е.П. Delphi для .NET.[Текст]/Е.П. Марков, В.В.Никифоров // СПб.: БХВ- Петербург, 2015. - 896 с : ил.

- 27 Матросов, А.В. HTML 4.0. [Текст] / А.В. Матросов, А.О. Сергеев, М.П. Чаунин // СПб.: БХВ-Петербург, 2013. - 672 с.: ил.
- 28 Милов, А.В. Основы программирования. [Текст] / А.В. Милов // М.: ООО Издательство «АСТ», 2010. - 397 с.
- 29 Михеева, Е.В. Информационные технологии в профессиональной деятельности [Электронный ресурс] / Е.В. Михеева // 7-е изд. – М.: Издательский центр «Академия», 2010г.
- 30 Праг, И. Access 2010. Библия пользователя. [Текст] И. Праг // М.: Издательский дом «Вильямс», 2012. – 1040с.
- 31 Практикум по информационным технологиям в профессиональной деятельности [Электронный ресурс] - 7-е изд. – М.: Издательский центр «Академия», 2011г.
- 32 Ульман, Д. «Теория и практика БД». [Текст] / Д. Ульман // М.: Машиностроение, 2013 г.-250 с.
- 33 Фаронов, В.В. Программирование баз данных в Delphi 7. Учебный курс. [Текст] / В.В. Фирюров // СПб.: Питер - 2010. – 459 с.: ил.
- 34 Харитонова, И.А. Microsoft Access 2010: разработка приложений. [Текст] / И.А. Харитонова, В.Д. Михеева // СПб.: БХВ – Петербург 2012. – 832с.
- 35 Шевченко, Н.А. Access 2010. Искусствосоздания базы данных [Текст] / Н.А. Шевченко // М.: ИТ Пресс, 2010. – 160 с.: ил.

ПРИЛОЖЕНИЯ

Приложение А

Внешний вид среды быстрой разработки приложений CodeGearRADStudio 2007



Приложение Б

Система управления базами данных MicrosoftOfficeAccess 2007



Приложение В

Таблица В.1 – Назначение и свойства объектов

Наименование объекта	Назначение объекта	Свойства объекта
1	2	3
ADODConnection1	Объект соединения	ConnectionString-Provider=MSDASQL.1;Persist Info=False;Data Source=СЭС; Connected = True LoginPrompt = True Security
Форма FOtdeli TADODDataSet1	Набор данных	Active=True Connection=Form8->ADODConnection1 CommandText = 'select * from Otdeli'
Форма FDolgnost TADODDataSet1	Набор данных	Active=True Connection= Form8->ADODConnection1 CommandText = 'select* from Dolgnosti'
Форма FObrazov TADODDataSet1	Набор данных	Active=True Connection= Form8->ADODConnection1 CommandText = 'select * from Obrazov'
Форма FSotrudniki TADODDataSet1	Набор данных	Active=True Connection= ADODConnection1 CommandTextSELECT Sotrudniki.Kod_sotrudnika, Sotrudniki.Kod_dolgnosti, Sotrudniki.Kod_obrazovaniya, Sotrudniki.Kod_otdela, Sotrudniki.Familiya, Sotrudniki.Imya, Sotrudniki.Otchestvo, Sotrudniki.Specialnost, Sotrudniki.Stag_raboti, Sotrudniki.Stag_raboti_pospec, Sotrudniki.Cem_pologenie, Sotrudniki.Data_rogdeniya, Sotrudniki.Vozrast, Sotrudniki.Pol, Sotrudniki.Nacionalnost, Sotrudniki.Adrec_progivaniya, Dolgnosti.Nazv_dolgnosti, Otdeli.Nazv_otdela, Obrazovanie.Vid_obrazovaniya FROM Obrazovanie INNER JOIN (Otdeli INNER JOIN (Dolgnosti INNER JOIN Sotrudniki ON Dolgnosti.[Kod_dolgnosti] = Sotrudniki.[Kod_dolgnosti]) ON

Продолжение Таблицы В.1– Назначение и свойства объектов

Наименование объекта	Назначение объекта	Свойства объекта
1	2	3
		Otdeli.[Kod_otdela] = Sotrudniki.[Kod_otdela]) ONObrazovanie.Kod_obrazovaniya = Sotrudniki.Kod_obrazovaniya;
Форма FSotrudniki TADOCCommand1	Команда реализации SQL запроса	Connection= ADOConnection1 CommandText = InsertintoSotrudniki (Kod_sotrudnika, Kod_dolgnosti, Kod_obrazovaniya, Kod_otdela, Familiya, Imya, Otchestvo, Specialnost, Stag_raboti, Stag_raboti_pospec, Cem_pologenie, Data_rogdeniya, Vozrast, Pol, Nacionalnost, Adrec_progivaniya)values (:Kod_sotrudnika,:Kod_dolgnosti,:Kod_obraz ovaniya,:Kod_otdela,:Familiya,:Imya,:Otchest vo,:Specialnost,:Stag_raboti,:Stag_raboti_pos pec,:Cem_pologenie,:Data_rogdeniya,:Vozras t,:Pol,:Nacionalnost,:Adrec_progivaniya)
Форма FSotrudniki TADOCCommand2	Команда реализации SQL запроса	Connection=ADOConnection1 CommandText =‘Update Sotrudniki Set Kod_dolgnosti=:Kod_dolgnosti, Kod_obrazovaniya=:Kod_obrazovaniya, Kod_otdela=:Kod_otdela, Familiya=:Familiya, Imya=:Imya, Otchestvo=:Otchestvo, Specialnost=:Specialnost, Stag_raboti=:Stag_raboti, Stag_raboti_pospec=:Stag_raboti_pospec, Cem_pologenie=:Cem_pologenie, Data_rogdeniya=:Data_rogdeniya, Vozrast=:Vozrast, Pol=:Pol, Nacionalnost=:Nacionalnost, Adrec_progivaniya=:Adrec_progivaniya where Kod_sotrudnika=:Kod_sotrudnika’
Форма FSotrudniki	Команда реализации	Connection= ADOConnection1 CommandText =‘Delete from Sotrudniki

Продолжение Таблицы В.1– Назначение и свойства объектов

Наименование объекта	Назначение объекта	Свойства объекта
1	2	3
TADOCCommand3	SQL запроса	where Kod_sotrudnika=:Kod_sotrudnika’
Форма FDolgnosti TADOCCommand1	Команда реализации SQL запроса	Connection= Form8.ADOConnection1 CommandText = ‘Insert into Dolgnosti (Kod_dolgnosti, Nazv_dolgnosti, Vipolnyamie_funkcii) values (:Kod_dolgnosti,:Nazv_dolgnosti,:Vipolnyam ie_funkcii) ’
Форма FDolgnosti TADOCCommand2	Команда реализации SQL запроса	Connection=Form8.ADOConnection1 CommandText=‘Update Dolgnosti set Nazv_dolgnosti=:Nazv_dolgnosti, Vipolnyamie_funkcii=:Vipolnyamie_funkcii where Kod_dolgnosti=:Kod_dolgnosti‘
Форма FOborud TADOCCommand1	Команда реализации SQL запроса	Connection=Form8.ADOConnection1 CommandText = ‘Insert into Oborudovanie (Kod_oborudovaniya, Naim_oborudovaniya, Izgotovitel, Data_ekscpluatacii, Teh_harakteristiki, Dokumentaciya, Kod_sotrudnika) values (:Kod_oborudovaniya,:Naim_oborudovaniya,: Izgotovitel,:Data_ekscpluatacii,:Teh_haracteri stiki,:Dokumentaciya,:Kod_sotrudnika)’
Форма FOborud TADOCCommand2	Команда реализации SQL запроса	Connection=Form8.ADOConnection1 CommandText = ‘Update Oborudovanie set Naim_oborudovaniya=:Naim_oborudovaniya, Izgotovitel=:Izgotovitel, Data_ekscpluatacii=:Data_ekscpluatacii, Teh_harakteristiki=:Teh_harakteristiki, Dokumentaciya=:Dokumentaciya, Kod_sotrudnika=:Kod_sotrudnika where Kod_oborudovaniya=:Kod_oborudovaniya’

Продолжение Таблицы В.1– Назначение и свойства объектов

Наименование объекта	Назначение объекта	Свойства объекта
1	2	3
Форма FOborud TADOCCommand3	Команда реализации SQL запроса	Connection=Form8.ADOConnection1 CommandText = 'Delete from Oborudovanie where Kod_oborudovaniya=:Kod_oborudovaniya'
DBGrid1	Вывод на экран данных из таблицы	DataSource=DataSource1
ToolBar1	Панель инструментов для кнопок	Images: FSotrudniki.ImageList1 Flat: true
ImageList1	Для хранения изображений	по умолчанию
ToolButton1	Кнопка на панели ToolBar для вызова необходимой функции	Caption: подпись кнопки ImageIndex: индекс изображения, которое хранится в ImageList Hint: всплывающая надпись ShowHint: Active
WebBrowser1	Для отображения HTML-страниц	по умолчанию
Edit1	Для ввода текста с клавиатуры	Caption: текст
Image1	Для загрузки изображения	Picture: выбор картинки
Label1	Для вывода текста	Caption: текст компонента
CheckBox1	Для установки флажка, да/нет	по умолчанию
Button1	Кнопка для вызова необходимой функции	Caption: надпись кнопки
MainMenu1	Для создания меню	Bitmap: вставка изображения Caption: названия пунктов меню и их подразделов

Приложение Г

Таблица Г.1– Структура таблицы «Sotrudniki»

Наименование поля	Тип данных	Ключ
Kod_sotrudnika	Числовой	первичный ключ
Kod_dolgnosti	Числовой	внешний ключ
Kod_obrazovaniya	Числовой	внешний ключ
Kod_otdela	Числовой	внешний ключ
Familiya	Текстовый	
Imya	Текстовый	
Otchestvo	Текстовый	
Specialnost	Текстовый	
Stag_raboti	Числовой	
Stag_raboti_pospec	Числовой	
Cem_pologenie	Текстовый	
Data_rogdeniya	Дата/Время	
Vozrast	Числовой	
Pol	Текстовый	
Nacionalnost	Текстовый	
Adrec_progivaniya	Текстовый	

Таблица Г.2– структура таблицы «Dolgnosti»

Наименование поля	Тип данных	Ключ
Kod_dolgnosti	Числовой	первичный ключ
Nazv_dolgnosti	Текстовый	
Vipolnyamie_funkcii	Текстовый	

Таблица Г.3– структура таблицы «Oborudovanie»

Наименование поля	Тип данных	Ключ
Kod_oborudovaniya	Числовой	первичный ключ
Naim_oborudovaniya	Текстовый	
Izgotovitel	Текстовый	
Data_ekscpluatacii	Числовой	
Teh_harakteristiki	Текстовый	
Dokumentaciya	Текстовый	
Kod_sotrudnika	Числовой	внешний ключ

Таблица Г.4– структура таблицы «Образование»

Наименование поля	Тип данных	Ключ
Kod_obrazovaniya	Числовой	первичный ключ
Vid_obrazovaniya	Текстовый	
Nazv_uchebnogo_zavedeniya	Текстовый	
Nazv_otdeleniya	Текстовый	

Таблица Г.5– структура таблицы «Otdeli»

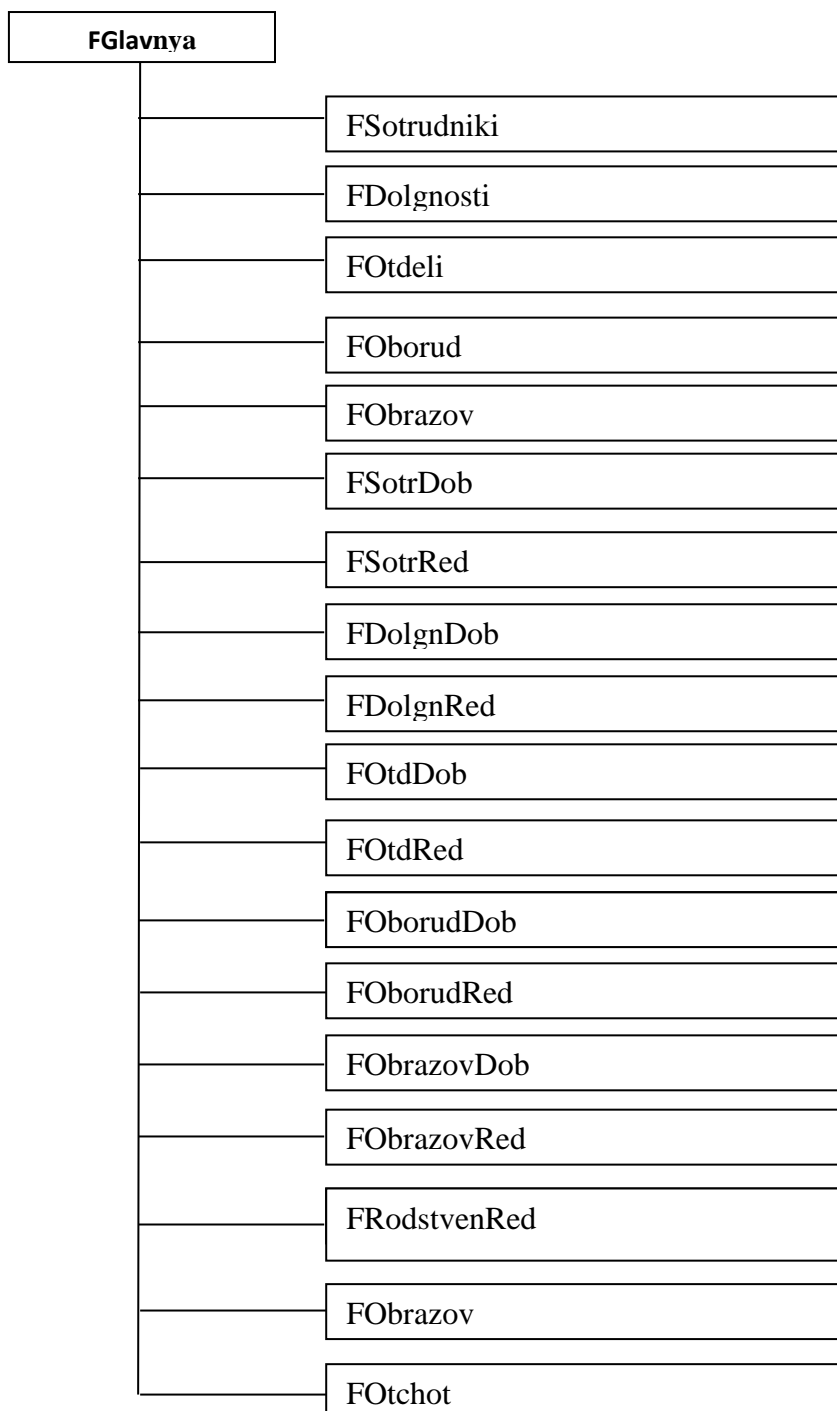
Наименование поля	Тип данных	Ключ
Kod_otdela	Числовой	первичный ключ
Nazv_otdela	Текстовый	

Таблица Г.6– Структура таблицы «Svedenia_o_rodstvennikah»

Наименование поля	Тип данных	Ключ
Kod_rodstvennika	Числовой	первичный ключ
Kod_sotrudnika	Числовой	внешний ключ
Familiya	Текстовый	
Imya	Текстовый	
Otchestvo	Текстовый	
Data_rogd	Дата/Время	
Stepen_rodstva	Текстовый	
Adres_rodstvennika	Текстовый	

Приложение Д

Модульная структура программного приложения



Приложение Е

Листинг программы

```
procedure TForm8.Button1Click(Sender: TObject);
begin
  KeyFields:='Kod_sotrudnika';
  KeyValues:=edit2.text;
  ResultFields:='Kod_sotrudnika;Familiya;Imya;Otchestvo;Specialnost;Stag_raboti;Stag_raboti_pospec;Cem_pologenie
;Data_rogdeniya;Voзраст;Pol;Nacionalnost;Adrec_progivaniya;Nazv_dolgnosti;Nazv_otdela;Vid_obrazovaniya';
  vrntResult:=Adodataset1.Lookup(KeyFields,KeyValues,ResultFields);
  if not VarIsNull(vrntResult) then
  begin
    if not VarIsNull(vrntResult[0]) then
      strResult:='Кодсотрудника: '+string(vrntResult[0])
    else strResult:='Kod_sotrudnika: -';
    if not VarIsNull(vrntResult[1]) then
      strResult:=strResult+#13#10+'Фамилия: '+string(vrntResult[1])
    else strResult:=strResult+#13#10+'Familiya: -';
    if not VarIsNull(vrntResult[2]) then
      strResult:=strResult+#13#10+'Имя: '+string(vrntResult[2])
    else strResult:=strResult+#13#10+'Imya: -';
    if not VarIsNull(vrntResult[3]) then
      strResult:=strResult+#13#10+'Отчество: '+string(vrntResult[3])
    else strResult:=strResult+#13#10+'Otchestvo: -';
    if not VarIsNull(vrntResult[4]) then
      strResult:=strResult+#13#10+'Специальность: '+string(vrntResult[4])
    else strResult:=strResult+#13#10+'Specialnost: -';
    if not VarIsNull(vrntResult[5]) then
      strResult:=strResult+#13#10+'Стажработы: '+string(vrntResult[5])
    else strResult:=strResult+#13#10+'Stag_raboti: -';
    if not VarIsNull(vrntResult[6]) then
      strResult:=strResult+#13#10+'Стажработыпоспециальности: '+string(vrntResult[6])
    else strResult:=strResult+#13#10+'Stag_raboti_pospec: -';
    if not VarIsNull(vrntResult[7]) then
      strResult:=strResult+#13#10+'Семейноеположение: '+string(vrntResult[7])
    else strResult:=strResult+#13#10+'Cem_pologenie: -';
    if not VarIsNull(vrntResult[8]) then
      strResult:=strResult+#13#10+'Датарождения: '+string(vrntResult[8])
    else strResult:=strResult+#13#10+'Data_rogdeniya: -';
    if not VarIsNull(vrntResult[9]) then
      strResult:=strResult+#13#10+'Возраст: '+string(vrntResult[9])
    else strResult:=strResult+#13#10+'Voзраст: -';
    if not VarIsNull(vrntResult[10]) then
      strResult:=strResult+#13#10+'Пол: '+string(vrntResult[10])
    else strResult:=strResult+#13#10+'Pol: -';
    if not VarIsNull(vrntResult[11]) then
      strResult:=strResult+#13#10+'Национальность: '+string(vrntResult[11])
    else strResult:=strResult+#13#10+'Nacionalnost: -';
    if not VarIsNull(vrntResult[12]) then
      strResult:=strResult+#13#10+'Адреспроживания: '+string(vrntResult[12])
    else strResult:=strResult+#13#10+'Adrec_progivaniya: -';
    if not VarIsNull(vrntResult[13]) then
      strResult:=strResult+#13#10+'Название: '+string(vrntResult[13])
    else strResult:=strResult+#13#10+'Nazv_dolgnosti: -';
    if not VarIsNull(vrntResult[14]) then
      strResult:=strResult+#13#10+'Названиеотдела: '+string(vrntResult[14])
    else strResult:=strResult+#13#10+'Nazv_otdela: -';
    MessageDLG('Данныенайденнойзаписи'+#13#10+strResult,mtInformation,[mbOK],0);
  else begin
```

```

Beep;
MessageDLG('Записьненайдена',mtInformation,[mbOK],0);
end;
end;
procedure TForm8.DBGrid1TitleClick(Column: TColumn);
begin
if column.Index=1then
begin
A:='Familiya';
B:='Familiya';
label1.Caption:= 'фамилии';
end;
if column.Index=4then
begin
A:='Specialnost';
B:='Specialnost';
label1.Caption:= 'специальности';
end;
if column.Index=13then
begin
A:='Nazv_dolgnosti';
B:='Nazv_dolgnosti';
label1.Caption:= 'названию должности';
end;
begin
if column.Index=14then
begin
A:='Nazv_otdela';
B:='Nazv_otdela';
label1.Caption:= 'названиюотдела';
end;
begin
if column.Index=15then
begin
A:='Vid_obrazovaniya';
B:='Vid_obrazovaniya';
label1.Caption:= 'видуобразования';
end;
procedure TForm8.Edit1Change(Sender: TObject);
begin
if edit1.Text=" thenadodataset1.Filtered:=false
else
begin
adodataset1.Filtered:=false;
adodataset1.Filter:=a + ' LIKE ' + #39 +
edit1.Text + '%' + #39;
adodataset1.Filtered:=true;
end;
procedure TForm8.N11Click(Sender: TObject);
begin
HTML:=TSTRINGLIST.Create;
HTML.CLEAR;
HTML.ADD('<HTML><TITLE>Сотрудники</TITLE>');
HTML.ADD('<BODY BGCOLOR="WHITE"><P ALIGN=CENTER><FONT SIZE="5">Сотрудники</P>');
HTML.ADD('<BODY BGCOLOR="WHITE"><P ALIGN=CENTER><FONT
SIZE="4">"Центрагигиеныиэпидемиологии" БелгородскойобластивНовооскольскомрайоне</P>');
HTML.ADD ('<P ALIgn=center>посостояниюна '+' <u><i>'+FormatDateTime('dd mmmm yyyy',now) +
'<br></i></u></P>');HTML.ADD('<TABLE WIDTH=100% BORDER"2" ALIGN="CENTER" BORDER
COLOR="BLACK">');
HTML.ADD('<TR>');
HTML.ADD('<TD WIDTH=15% ALIGN=CENTER><FONT SIZE="3"><B>Названиедолжности</b></td>');
HTML.ADD('<TD WIDTH=15% ALIGN=CENTER><FONT SIZE="3"><B>Видобразования</b></td>');
HTML.ADD('<TD WIDTH=15% ALIGN=CENTER><FONT SIZE="3"><B>Названиеотдела</b></td>');

```



```

Form28.show;
end;
procedure TForm8.N14Click(Sender: TObject);
begin
Form29.show;
end;
procedure TForm8.N15Click(Sender: TObject);
begin
Form30.show;
end;
procedure TForm8.N16Click(Sender: TObject);
begin
Form8.ADODataSet1.Filtered:=false;
end;
procedure TForm8.N17Click(Sender: TObject);
begin
Form34.Caption:='Фильтрация сотрудников по дате рождения';
if Form34.ShowModal=mrOk then
begin
ADODataSet1.Filter:='';
ADODataSet1.Filtered:=true;
ADODataSet1.Filter:='Data_rogdeniya>=' + DateToStr(Form34.DateTimePicker1.Date) +
' and Data_rogdeniya<=' + DateToStr(Form34.DateTimePicker2.Date);
end;
end;
procedure TForm8.N18Click(Sender: TObject);
begin
adodataset1.Filtered:=false;
end;
procedure TForm8.N20Click(Sender: TObject);
begin
Form31.Show;
end;
procedure TForm8.N21Click(Sender: TObject);
begin
HTML:=TSTRINGLIST.Create;
HTML.CLEAR;
HTML.ADD('<HTML><TITLE>Сотрудники</TITLE>');
HTML.ADD('<BODY BGCOLOR="WHITE"><P ALIGN=CENTER><FONT SIZE="5">Сотрудники</P>');
HTML.ADD('<BODY BGCOLOR="WHITE"><P ALIGN=CENTER><FONT
SIZE="4">"Центра гигиены и эпидемиологии" Белгородской области в Новооскольском районе</P>');
HTML.ADD('<P ALIGN=center>посостояния на '+' + FormatDateTime('dd mmmm yyyy',now) +
'<br></i></u></P>');HTML.ADD('<TABLE WIDTH=100% BORDER"2" ALIGN="CENTER" BORDER
COLOR="BLACK">');
HTML.ADD('<TR>');
HTML.ADD('<TD WIDTH=15% ALIGN=CENTER><FONT SIZE="3"><B>Фамилия</b></td>');
HTML.ADD('<TD WIDTH=15% ALIGN=CENTER><FONT SIZE="3"><B>Имя</b></td>');
HTML.ADD('<TD WIDTH=15% ALIGN=CENTER><FONT SIZE="3"><B>Отчество</b></td>');
HTML.ADD('<TD WIDTH=15% ALIGN=CENTER><FONT SIZE="3"><B>Специальность</b></td>');
HTML.ADD('<TD WIDTH=15% ALIGN=CENTER><FONT SIZE="3"><B>Стаж работы</b></td>');
HTML.ADD('<TD WIDTH=15% ALIGN=CENTER><FONT
SIZE="3"><B>Стаж работы по специальности</b></td>');
HTML.ADD('<TD WIDTH=15% ALIGN=CENTER><FONT SIZE="3"><B>Семейное положение</b></td>');
HTML.ADD('<TD WIDTH=15% ALIGN=CENTER><FONT SIZE="3"><B>Дата рождения</b></td>');
HTML.ADD('<TD WIDTH=15% ALIGN=CENTER><FONT SIZE="3"><B>Возраст</b></td>');
HTML.ADD('<TD WIDTH=15% ALIGN=CENTER><FONT SIZE="3"><B>Национальность</b></td>');
HTML.ADD('<TD WIDTH=15% ALIGN=CENTER><FONT SIZE="3"><B>Пол</b></td>');
HTML.ADD('<TD WIDTH=15% ALIGN=CENTER><FONT SIZE="3"><B>Адрес проживания</b></td>');
HTML.ADD('<TD WIDTH=15% ALIGN=CENTER><FONT SIZE="3"><B>Примечание</b></td>');
ADODATASET2.First;
for I:=1 TO ADODATASET2.RECORDCOUNT DO
BEGIN
HTML.ADD('<TR>');;

```

```

HTML.ADD('<TD WIDTH=20% ALIGN=CENTER>'+ADODataSet2Familiya.ASSTRING+'</TD>');
HTML.ADD('<TD WIDTH=20% ALIGN=CENTER>'+ADODataSet2Imya.ASSTRING+'</TD>');
HTML.ADD('<TD WIDTH=20% ALIGN=CENTER>'+ADODataSet2Ochestvo.ASSTRING+'</TD>');
HTML.ADD('<TD WIDTH=20% ALIGN=CENTER>'+ADODataSet2Specialnost.ASSTRING+'</TD>');
HTML.ADD('<TD WIDTH=20% ALIGN=CENTER>'+ADODataSet2Stag_raboti.ASSTRING+'</TD>');
HTML.ADD('<TD WIDTH=20% ALIGN=CENTER>'+ADODataSet2Stag_raboti_pospec.ASSTRING+'</TD>');
HTML.ADD('<TD WIDTH=20% ALIGN=CENTER>'+ADODataSet2Cem_pologenie.ASSTRING+'</TD>');
HTML.ADD('<TD WIDTH=20% ALIGN=CENTER>'+ADODataSet2Data_rogdeniya.ASSTRING+'</TD>');
HTML.ADD('<TD WIDTH=20% ALIGN=CENTER>'+ADODataSet2Voзраст.ASSTRING+'</TD>');
HTML.ADD('<TD WIDTH=20% ALIGN=CENTER>'+ADODataSet2Nacionalnost.ASSTRING+'</TD>');
HTML.ADD('<TD WIDTH=20% ALIGN=CENTER>'+ADODataSet2Pol.ASSTRING+'</TD>');
HTML.ADD('<TD WIDTH=20% ALIGN=CENTER>'+ADODataSet2Adrec_progivaniya.ASSTRING+'</TD>');
HTML.ADD('<TD WIDTH=20% ALIGN=CENTER>'+ADODataSet2Primechanie.ASSTRING+'</TD>');
HTML.ADD('</TR>');
ADODATASET1.NEXT;
END;
HTML.ADD('</TABLE>');
html.Add('<table width=100% border="0">');
html.Add('<tr>');
html.Add('<td width="410" scope="col"><FONT SIZE=4><div align="left"><span
class="стиль2">Начальник</span></div></FONT></td>');
html.Add('<th width="362" scope="col">&nbsp;</th>');
html.Add('</tr>');
html.Add('<tr>');
html.Add('<td scope="col"><FONT SIZE=4><div align="left"
class="стиль2">отдела кадров:</div></FONT></td>');
html.Add('<td width="362" scope="col"><FONT SIZE=4><div align="right"><span
class="стиль2">/_____/_____/</span></div></FONT></td>');
html.Add('</tr>');
html.Add('</table>');
HTML.ADD('</BODY></HTML>');
FORM20.Caption:='Сотрудники';
filename:= 'Сотрудники';
html.savetofile(filename+'.htm');
getdir(0,path);
form20.Webbrowser1.Navigate(path+'\'+filename+'.htm');
form20.Show;
end;
procedure TForm8.N2Click(Sender: TObject);
begin
Form8.Close;
end
procedure TForm8.N5Click(Sender: TObject);
begin
Form1.ShowModal;
end;
procedure TForm8.N6Click(Sender: TObject);
begin
Form6.Show;
end;
procedure TForm8.N7Click(Sender: TObject);
begin
Form11.Show;
end;
procedure TForm8.N8Click(Sender: TObject);
begin
Form14.Show;
end;
procedure TForm8.N9Click(Sender: TObject);
begin
Form17.Show;
end;
procedure TForm8.ToolButton10Click(Sender: TObject);

```

```

begin
Form8.ADODataSet1.Filtered:=false;
end;
procedure TForm8.ToolButton11Click(Sender: TObject);
begin
begin
Form32.ADODataSet1.Close;
Form32.ADODataSet1.Parameters.ParamByName('Kod_sotrudnika').Value:=ADODataSet1Kod_sotrudnika.AsString
;
Form32.ADODataSet1.Open;
Form32.Show;
end;
procedure TForm8.ToolButton12Click(Sender: TObject);
begin
adodataSet1.Filtered:=false;
end;
procedure TForm8.ToolButton14Click(Sender: TObject);
begin
Form33.ADODataSet1.Close;
Form33.ADODataSet1.Parameters.ParamByName('Kod_sotrudnika').Value:=ADODataSet1Kod_sotrudnika.AsString
;
Form33.ADODataSet1.Open;
Form33.Show;
end;
procedure TForm8.ToolButton15Click(Sender: TObject);
begin
begin
Form34.Caption:='Фильтрация сотрудников по дате рождения';
if Form34.ShowModal=mrOk then
begin
ADODataSet1.Filter:='';
ADODataSet1.Filtered:=true;
ADODataSet1.Filter:= 'Data_rogdeniya>=' + DateToStr(Form34.DateTimePicker1.Date) +
' and Data_rogdeniya<=' + DateToStr(Form34.DateTimePicker2.Date);
end;
procedure TForm8.ToolButton16Click(Sender: TObject);
begin
begin
adodataset1.Filtered:=false;
end;
procedure TForm8.ToolButton1Click(Sender: TObject);
begin
if Form2. showModal=mrOk then
begin
Adocommand1.Parameters.ParamByName('Kod_sotrudnika').value:=Form2.Edit1.text;
Adocommand1.Parameters.ParamByName('Kod_dolgnosti').value:=Form6.ADODataSet1Kod_dolgnosti.asstring;;Ad
ocommand1.Parameters.ParamByName('Kod_obrazovaniya').value:=Form1.ADODataSet1Kod_obrazovaniya.AsString;
Adocommand1.Parameters.ParamByName('Kod_otdela').value:=Form11.ADODataSet1Kod_otdela.asstring;
Adocommand1.Parameters.ParamByName('Familiya').value:=Form2.Edit5.text;
Adocommand1.Parameters.ParamByName('Imya').value:=Form2.Edit6.text;
Adocommand1.Parameters.ParamByName('Otchestvo').value:=Form2.Edit7.text;
Adocommand1.Parameters.ParamByName('Specialnost').value:=Form2.Edit8.text;
Adocommand1.Parameters.ParamByName('Stag_raboti').value:=Form2.Edit9.text;Adocommand1.Parameters.Param
ByName('Stag_raboti_pospec').value:=Form2.Edit10.text;Adocommand1.Parameters.ParamByName('Cem_pologenie').value:=F
orm2.ComboBox1.text;Adocommand1.Parameters.ParamByName('Data_rogdeniya').value:=Form2.dateTimePicker1.Date;
Adocommand1.Parameters.ParamByName('Voзраст').value:=Form2.Edit13.text;
Adocommand1.Parameters.ParamByName('Pol').value:=Form2.ComboBox1.text;
Adocommand1.Parameters.ParamByName('Nacionalnost').value:=Form2.Edit15.text;
Adocommand1.Parameters.ParamByName('Adrec_progivaniya').value:=Form2.Edit16.text;
Adocommand1.Parameters.ParamByName('Adrec_progivaniya').value:=Form2.Edit16.text;
Adocommand1.Parameters.ParamByName('Primechanie').value:=Form2.ComboBox3.text;
AdoCommand1.Execute;
Adodataset1.Close;

```

```

Adodataset1.Open;
end;
procedure TForm8.ToolButton2Click(Sender: TObject);
begin
Form3.Edit1.text:=ADODataset1.Kod_sotrudnika.AsString;
Form3.Edit2.text:=ADODataset1.Kod_dolgnosti.AsString;
Form3.Edit3.text:=ADODataset1.Kod_obrazovaniya.AsString;
Form3.Edit4.text:=ADODataset1.Kod_otdela.AsString;
Form3.Edit5.text:=ADODataset1.Familiya.AsString;
Form3.Edit6.text:=ADODataset1.Imya.AsString;
Form3.Edit7.text:=ADODataset1.Otchestvo.AsString;
Form3.Edit8.text:=ADODataset1.Specialnost.AsString;
Form3.Edit9.text:=ADODataset1.Stag_raboti.AsString;
Form3.Edit10.text:=ADODataset1.Stag_raboti_pospec.AsString;
Form3.ComboBox2.text:=ADODataset1.Cem_pologenie.AsString;
Form3.DateTimePicker1.Date:=ADODataset1.Data_rogdeniya.AsDateTime;
Form3.Edit13.text:=ADODataset1.Vozrast.AsString;
Form3.ComboBox1.text:=ADODataset1.Pol.AsString;
Form3.Edit15.text:=ADODataset1.Nacionalnost.AsString;
Form3.Edit16.text:=ADODataset1.Adrec_progivaniya.AsString;
Form3.ComboBox3.text:=ADODataset1.Primechanie.AsString;
if Form3.ShowModal=mrok then
Begin
Adocommand2.Parameters.ParamByName('Kod_sotrudnika').value:=Form3.Edit1.text;
Adocommand2.Parameters.ParamByName('Kod_dolgnosti').value:=Form6.ADODataset1.Kod_dolgnosti.asstring;
Adocommand2.Parameters.ParamByName('Kod_obrazovaniya').value:=Form1.ADODataset1.Kod_obrazovaniya.As
String;
Adocommand2.Parameters.ParamByName('Kod_otdela').value:=ADODataset1.Kod_otdela.asstring;
Adocommand2.Parameters.ParamByName('Familiya').value:=Form3.Edit5.text;
Adocommand2.Parameters.ParamByName('Imya').value:=Form3.Edit6.text;
Adocommand2.Parameters.ParamByName('Otchestvo').value:=Form3.Edit7.text;
Adocommand2.Parameters.ParamByName('Specialnost').value:=Form3.Edit8.text;
Adocommand2.Parameters.ParamByName('Stag_raboti').value:=Form3.Edit9.text;
Adocommand2.Parameters.ParamByName('Stag_raboti_pospec').value:=Form3.Edit10.text;
Adocommand2.Parameters.ParamByName('Cem_pologenie').value:=Form3.ComboBox2.text;
Adocommand2.Parameters.ParamByName('Data_rogdeniya').value:=Form3.DateTimePicker1.Date;
Adocommand2.Parameters.ParamByName('Vozrast').value:=Form3.Edit13.text;
Adocommand2.Parameters.ParamByName('Pol').value:=Form3.ComboBox1.text;
Adocommand2.Parameters.ParamByName('Nacionalnost').value:=Form3.Edit15.text;
Adocommand2.Parameters.ParamByName('Adrec_progivaniya').value:=Form3.Edit16.text;Adocommand2.Parameter
s.ParamByName('Primechanie').value:=Form3.ComboBox3.text;
if MessageBox(Handle,'Сохранитьизменения?','Редактирование... ',MB_YESNO)=idYES then
begin
Adocommand2.execute;
AdoDataSet1.Close;
AdoDataSet1.Open
end;
End;
end;
procedure TForm8.ToolButton3Click(Sender: TObject);
begin
if MessageBox(Handle,'Удалитьзапись?','Удаление... ',MB_YESNO)=IDYES then
begin
Adocommand3.Parameters.ParamByName('Kod_sotrudnika').value:=ADODataset1.Kod_sotrudnika.AsString;
Adocommand3.Execute;
AdodataSet1.Close;
Adodataset1.Open;
end;
end;
procedure TForm8.ToolButton5Click(Sender: TObject);
begin
sort_direction:='ASC';
adodataSet1.Sort:=b+' '+sort_direction;

```

```

end;
procedure TForm8.ToolButton6Click(Sender: TObject);
begin
sort_direction:='DESC';
adodataSet1.Sort:=b+' '+sort_direction;
procedure TForm8.ToolButton7Click(Sender: TObject);
begin
adodataset1.Active:=false;
adodataset1.Active:=true;
sort_direction:='ASC';
adodataset1.Sort:='Kod_sotrudnika'+ ' '+sort_direction;
procedure TForm8.ToolButton8Click(Sender: TObject);
begin
HTML:=TSTRINGLIST.Create;
HTML.CLEAR;
HTML.ADD('<HTML><TITLE>Личный листок сотрудника</TITLE>');
HTML.ADD('<BODY>');
HTML.ADD('<P ALIGN=CENTER><FONT SIZE="5"><B>Личный листок сотрудника</b></FONT></P>');
HTML.ADD('<BODY BGCOLOR="WHITE"><P ALIGN=CENTER><FONT
SIZE="4">"Центра гигиены и эпидемиологии" Белгородской области в Новооскольском районе</P>');
HTML.ADD('<P ALign=center>посостояния юна '+'<u><i>'+FormatDateTime('dd mmmm yyyy',now) +
'<br></i></u></P>');
HTML.add('<OL>');
HTML.ADD('<LI><P><Font Size
=4>Фамилия_<U><B>'+ADODataset1.Familiya.ASSTRING+'</B></U></Font></P>');
HTML.ADD('<LI><P><Font Size =4>Имя_<U><B>'+ADODataset1.Imya.ASSTRING+'</B></U></Font></P>');
HTML.ADD('<LI><P><Font Size
=4>Отчество_<U><B>'+ADODataset1.Otchestvo.ASSTRING+'</B></U></Font></P>');
HTML.ADD('<LI><P><Font Size
=4>Специальность_<U><B>'+ADODataset1.Specialnost.ASSTRING+'</B></U></Font></P>');
HTML.ADD('<LI><P><Font Size
=4>Стаж работы_<U><B>'+ADODataset1.Stag_raboti.ASSTRING+'</B></U></Font></P>');
HTML.ADD('<LI><P><Font Size =4>Стаж работы по специальности
_<U><B>'+ADODataset1.Stag_raboti_pospec.ASSTRING+'</B></U></Font></P>');
HTML.ADD('<LI><P><Font Size
=4>Семейное положение_<U><B>'+ADODataset1.Cem_pologenie.ASSTRING+'</B></U></Font></P>');
HTML.ADD('<LI><P><Font Size
=4>Дата рождения_<U><B>'+ADODataset1.Data_rogdeniya.ASSTRING+'</B></U></Font></P>');
HTML.ADD('<LI><P><Font Size
4=>Возраст_<U><B>'+ADODataset1.Vozrast.ASSTRING+'</B></U></Font></P>');
HTML.ADD('<LI><P><Font Size
=4>Национальность_<U><B>'+ADODataset1.Nacionalnost.ASSTRING+'</B></U></Font></P>');
HTML.ADD('<LI><P><Font Size =4>Пол_<U><B>'+ADODataset1.Pol.ASSTRING+'</B></U></Font></P>');
HTML.ADD('<LI><P><Font Size
=4>Адрес проживания_<U><B>'+ADODataset1.Adrec_progivaniya.ASSTRING+'</B></U></Font></P>');
HTML.ADD('<LI><P><Font Size
=4>Вид образования_<U><B>'+ADODataset1.Vid_obrazovaniya.ASSTRING+'</B></U></Font></P>');
HTML.ADD('<LI><P><Font Size
=4>Название отдела_<U><B>'+ADODataset1.Nazv_otdela.ASSTRING+'</B></U></Font></P>');
HTML.ADD('<LI><P><Font Size =4>Название должности
_<U><B>'+ADODataset1.Nazv_dolgnosti.ASSTRING+'</B></U></Font></P>')
HTML.ADD('</TABLE>');
html.Add('<table width=100% border="0">');
html.Add('<tr>');
html.Add('<td width="410" scope="col"><FONT SIZE=4><div align="left"><span
class="стиль2">Начальник</span></div></FONT></td>');
html.Add('<th width="362" scope="col">&nbsp;</th>');
html.Add('</tr>');
html.Add('<tr>');
html.Add('<td scope="col"><FONT SIZE=4><div align="left"
class="стиль2">отдела кадров:</div></FONT></td>');
html.Add('<td width="362" scope="col"><FONT SIZE=4><div align="right"><span
class="стиль2">/_____/_____/_____/</span></div></FONT SIZE=4></td>');

```

```

html.Add('</tr>');
html.Add('</table>');
HTML.ADD('</BODY>');
HTML.ADD('</HTML>');
FORM20.Caption:='Личный листок сотрудника';
filename:='Личный листок сотрудника';
html.savetofile(filename+'.htm');
getdir(0,path);
form20.Webbrowser1.Navigate(path+'\'+filename+'.htm');
form20.Show;
procedure TForm1.DBGrid1TitleClick(Column: TColumn);
begin
A:='Vid_obrazovaniya';
B:='Vid_obrazovaniya';
end;
procedure TForm1.N2Click(Sender: TObject);
begin
Form1.Close;
end;
procedure TForm1.N4Click(Sender: TObject);
begin
Form8.Show;
end;
procedure TForm1.N5Click(Sender: TObject);
begin
Form17.Show;
end;
procedure TForm1.N6Click(Sender: TObject);
begin
Form6.Show;
end;
procedure TForm1.N7Click(Sender: TObject);
begin
Form11.Show;
end;
procedure TForm1.N8Click(Sender: TObject);
begin
Form14.Show;
end;
procedure TForm1.N9Click(Sender: TObject);
begin
HTML:=TSTRINGLIST.Create;
HTML.CLEAR;
HTML.ADD('<HTML><TITLE>Работники</TITLE>');
HTML.ADD('<BODY BGCOLOR="WHITE"><P ALIGN=CENTER><FONT
SIZE="6">Списокобразованиясотрудников</P>');
HTML.ADD('<BODY BGCOLOR="WHITE"><P ALIGN=CENTER><FONT SIZE="5">
"Центрагигиеныиэпидемиологии" БелгородскойобластиНовооскольскогорайона</P>');
HTML.ADD('<TABLE WIDTH=100% BORDER"2" ALIGN="CENTER" BORDER COLOR="BLACK">');
HTML.ADD('<TR>');
HTML.ADD('<TD WIDTH=15% ALIGN=CENTER><FONT SIZE="5"><B>Видобразования</b></td>');
HTML.ADD('<TD WIDTH=15% ALIGN=CENTER><FONT
SIZE="5"><B>Названиеучебногозаведения</b></td>');
HTML.ADD('<TD WIDTH=15% ALIGN=CENTER><FONT SIZE="5"><B>Названияотделения</b></td>');
for I:=1 TO ADODATASET1.RECORDCOUNT DO
BEGIN
HTML.ADD('<TR>');
HTML.ADD('<TD WIDTH=20%
ALIGN=CENTER>'+ADODataset1 Vid_obrazovaniya.ASSTRING+'</TD>');
HTML.ADD('<TD WIDTH=20%
ALIGN=CENTER>'+ADODataset1Nazv_ychebnogo_zavedeniya.ASSTRING+'</TD>');
HTML.ADD('<TD WIDTH=20% ALIGN=CENTER>'+ADODataset1Nazv_otdeleniya.ASSTRING+'</TD>');
HTML.ADD('</TR>');

```

```

ADODATASET1.NEXT;
END;
HTML.ADD('</TABLE>');
html.Add('<table width=100% border="0">');
html.Add('<tr>');
html.Add('<td width="410" scope="col"><FONT SIZE=4><div align="left"><span
class="стиль2">Начальник</span></div></FONT></td>');
html.Add('<th width="362" scope="col">&nbsp;   </th>');
html.Add('</tr>');
html.Add('<tr>');
html.Add('<td scope="col"><FONT SIZE=4><div align="left"
class="стиль2">отдела кадров:</div></FONT></td>');
html.Add('<td width="362" scope="col"><FONT SIZE=4><div align="right"><span
class="стиль2">/_____/_____/</span></div><FONT SIZE=4></td>');
html.Add('</tr>');
html.Add('</table>');
HTML.ADD('</BODY></HTML>');
FORM20.Caption:='Образование';
filename:='Список образования каждого сотрудника организации ';
html.savetofile(filename+'.htm');
getdir(0,path);
form20.Webbrowser1.Navigate(path+'\'+filename+'.htm');
form20.Show;
end;
procedure TForm1.ToolButton1Click(Sender: TObject);
begin
if Form4.showModal=mrOk then
begin
Adocommand1.Parameters.ParamByName('Kod_obrazovaniya').value:=Form4.Edit1.text;
Adocommand1.Parameters.ParamByName('Vid_obrazovaniya').value:=Form4.ComboBox1.text;
Adocommand1.Parameters.ParamByName('Nazv_ychebnogo_zavedeniya').value:=Form4.Edit3.text; Adocomma
nd1.Parameters.ParamByName('Nazv_otdeleniya').value:=Form4.Edit4.text
AdoCommand1.Execute;
Adodataset1.Close;
Adodataset1.Open;
end;
end;
procedure TForm1.ToolButton2Click(Sender: TObject);
begin
if MessageBox(Handle,'Удалить запись?', 'Удаление...', MB_YESNO)=IDYES then
begin
Adocommand3.Parameters.ParamByName('Kod_obrazovaniya').value:=ADODataset1Kod_obrazovaniya.asStri
ng;
Adocommand3.Execute;
AdodataSet1.Close;
Adodataset1.Open;
end;
end;
procedure TForm1.ToolButton3Click(Sender: TObject);
begin
Form5.Edit1.text:=ADODataset1Kod_obrazovaniya.AsString;
Form5.ComboBox1.text:=ADODataset1Vid_obrazovaniya.AsString;
Form5.Edit3.text:=ADODataset1Nazv_ychebnogo_zavedeniya.AsString;
Form5.Edit4.text:=ADODataset1Nazv_otdeleniya.AsString;
Begin
if Form5.ShowModal=mrOk then
Begin Adocommand2.Parameters.ParamByName('Kod_obrazovaniya').value:=Form5.Edit1.text;
Adocommand2.Parameters.ParamByName('Vid_obrazovaniya').value:=Form5.ComboBox1.text; Adocommand2.
Parameters.ParamByName('Nazv_ychebnogo_zavedeniya').value:=Form5.Edit3.text;
Adocommand2.Parameters.ParamByName('Nazv_otdeleniya').value:=Form5.Edit4.text;
if MessageBox(Handle,'Сохранить изменения?', 'Редактирование...', MB_YESNO)=idYES then
begin
Adocommand2.execute;

```



```

AdoDataSet1.Close;
AdoDataSet1.Open;
end;
End;
End;
end;
procedure TForm1.ToolButton5Click(Sender: TObject);
begin
sort_direction:='ASC';
adodataSet1.Sort:=b+' '+sort_direction;
procedure TForm1.ToolButton6Click(Sender: TObject);
begin
sort_direction:='DESC';
adodataSet1.Sort:=b+' '+sort_direction;
procedure TForm1.ToolButton7Click(Sender: TObject);
begin
adodataset1.Active:=false;
adodataset1.Active:=true;
sort_direction:='ASC';
adodataset1.Sort:='Kod_obrazovaniya'+ ' '+sort_direction;
procedure TForm1.ToolButton8Click(Sender: TObject);
begin
Rvsystem1.DefaultDest:=rdpreview;
rvproject1.execute;
end;
procedure TForm1.ToolButton9Click(Sender: TObject);
begin
Rvsystem1.DefaultDest:=rdpreview;
rvproject2.execute;
end;
begin
A:='Familiya';
B:='Familiya';
end;
end;
procedure TForm17.N2Click(Sender: TObject);
begin
Form17.Close;
end;
procedure TForm17.N4Click(Sender: TObject);
begin
Form8.Show;
end;
procedure TForm17.N5Click(Sender: TObject);
begin
Form14.Show;
end;
procedure TForm17.N6Click(Sender: TObject);
begin
Form1.Show;
end;
procedure TForm17.N7Click(Sender: TObject);
begin
Form6.Show;
end;
procedure TForm17.N8Click(Sender: TObject);
begin
Form11.Show;
end;
procedure TForm17.N9Click(Sender: TObject);
begin
HTML:=TSTRINGLIST.Create;
HTML.CLEAR;

```

```

HTML.ADD('<HTML><TITLE>Родственники</TITLE>');
HTML.ADD('<BODY BGCOLOR="WHITE"><P ALIGN=CENTER><FONT
SIZE="5">Список родственников сотрудника</P>');
HTML.ADD('<BODY BGCOLOR="WHITE"><P ALIGN=CENTER><FONT
SIZE="4">"Центра гигиены и эпидемиологии" Белгородской области Новооскольского района</P>');
HTML.ADD('<TABLE WIDTH=100% BORDER="2" ALIGN="CENTER" BORDER COLOR="BLACK">');
HTML.ADD('<TR>');
HTML.ADD('<TD WIDTH=15% ALIGN=CENTER><FONT
SIZE="4"><B>Фамилия сотрудника</b></td>');
HTML.ADD('<TD WIDTH=15% ALIGN=CENTER><FONT SIZE="4"><B>Фамилия</b></td>');
HTML.ADD('<TD WIDTH=15% ALIGN=CENTER><FONT SIZE="4"><B>Имя</b></td>');
HTML.ADD('<TD WIDTH=15% ALIGN=CENTER><FONT SIZE="4"><B>Отчество</b></td>');
HTML.ADD('<TD WIDTH=15% ALIGN=CENTER><FONT SIZE="4"><B>Дата рождения</b></td>');
HTML.ADD('<TD WIDTH=15% ALIGN=CENTER><FONT SIZE="4"><B>Степень родства</b></td>');
HTML.ADD('<TD WIDTH=15% ALIGN=CENTER><FONT SIZE="4"><B>Адрес проживания</b></td>');
ADODATASET1.First;
for I:=1 TO ADODATASET1.RECORDCOUNT DO
BEGIN
HTML.ADD('<TR>');
HTML.ADD('<TD WIDTH=20%
ALIGN=CENTER>'+ADODataset1.Sotrudniki_Familiya.ASSTRING+'</TD>');
HTML.ADD('<TD WIDTH=20%
ALIGN=CENTER>'+ADODataset1.Svedenia_o_rodstvennikah_Familiya.ASSTRING+'</TD>');
HTML.ADD('<TD WIDTH=20% ALIGN=CENTER>'+ADODataset1.Imya.ASSTRING+'</TD>');
HTML.ADD('<TD WIDTH=20% ALIGN=CENTER>'+ADODataset1.Otchestvo.ASSTRING+'</TD>');
HTML.ADD('<TD WIDTH=20% ALIGN=CENTER>'+ADODataset1.Data_rogd.ASSTRING+'</TD>');
HTML.ADD('<TD WIDTH=20% ALIGN=CENTER>'+ADODataset1.Stepen_rodstva.ASSTRING+'</TD>');
ALIGN=CENTER>'+ADODataset1.Adres_rodstvennika.ASSTRING+'</TD>');
HTML.ADD('</TR>');
ADODATASET1.NEXT;
END;
HTML.ADD('</TABLE>');
html.Add('<table width=100% border="0">');
html.Add('<tr>');
html.Add('<td width="410" scope="col"><FONT SIZE=4><div align="left"><span
class="стиль2">Начальник</span></div></FONT></td>');
html.Add('<th width="362" scope="col">&nbsp;</th>');
html.Add('<tr>');
html.Add('<tr>');
html.Add('<td scope="col"><FONT SIZE=4><div align="left"
class="стиль2">отдела кадров:</div></FONT></td>');
html.Add('<td width="362" scope="col"><FONT SIZE=4><div align="right"><span
class="стиль2">/_____/_____/</span></div></FONT></td>');
html.Add('</tr>');
html.Add('</table>');
HTML.ADD('</BODY></HTML>');
FORM20.Caption:='Родственники сотрудников';
filename:='Родственники сотрудников';
html.savetofile(filename+'.htm');
getdir(0,path);
form20.Webbrowser1.Navigate(path+''+filename+'.htm');
form20.Show;
end;
procedure TForm17.ToolButton1Click(Sender: TObject);
begin
if Form18.showModal=mrOk then
begin
Adocommand1.Parameters.ParamByName('Kod_rodstvennika').value:=Form18.Edit1.text;
Adocommand1.Parameters.ParamByName('Kod_sotrudnika').value:=Form21.ADODataset1.Kod_sotrudnika.ass
tring;Adocommand1.Parameters.ParamByName('Familiya').value:=Form18.Edit3.Text;
Adocommand1.Parameters.ParamByName('Imya').value:=Form18.Edit4.text;Adocommand1.Parameters.Param
ByName('Otchestvo').value:=Form18.Edit5.text;Adocommand1.Parameters.ParamByName('Data_rogd').value:=Form18.da

```

```

teTimePicker1.Date;Adocommand1.Parameters.ParamByName('Stepen_rodstva').value:=Form18.ComboBox1.text;
Adocommand1.Parameters.ParamByName('Adres_rodstvennika').value:=Form18.Edit8.text;
    AdoCommand1.Execute;
    Adodataset1.Close;
    Adodataset1.Open;
end;
procedure TForm17.ToolButton2Click(Sender: TObject);
begin
if MessageBox(Handle,'Удалитьзапись?','Удаление...',MB_YESNO)=IDYES then
begin
Adocommand3.Parameters.ParamByName('Kod_rodstvennika').value:=ADODataset1Kod_rodstvennika.asStrin
g;
    Adocommand3.Execute;
    AdodataSet1.Close;
    Adodataset1.Open;
end;
procedure TForm17.ToolButton3Click(Sender: TObject);
begin
Form19.Edit1.text:=ADODataset1Kod_rodstvennika.AsString;
Form19.Edit2.text:=ADODataset1Kod_sotrudnika.AsString;
Form19.Edit3.text:=ADODataset1Svedenia_o_rodstvennikah_Familiya.AsString;
Form19.Edit4.text:=ADODataset1Imya.AsString;
Form19.Edit5.text:=ADODataset1Otchestvo.AsString;
Form19.DateTimePicker1.Date:=ADODataset1Data_rogd.AsdateTime;
Form19.ComboBox1.text:=ADODataset1Stepen_rodstva.AsString;
Form19.Edit8.text:=ADODataset1Adres_rodstvennika.AsString;
Begin
if Form19.ShowModal=mrok then
Begin
Adocommand2.Parameters.ParamByName('Kod_rodstvennika').value:=Form19.Edit1.text;
Adocommand2.Parameters.ParamByName('Kod_sotrudnika').value:=Form21.ADODataset1Kod_sotrudnika.asStrin
g;Adocommand2.Parameters.ParamByName('Familiya').value:=Form19.Edit3.Text;
Adocommand2.Parameters.ParamByName('Imya').value:=Form19.Edit4.text;Adocommand2.Parameters.Param
ByName('Otchestvo').value:=Form19.Edit5.text;Adocommand2.Parameters.ParamByName('Data_rogd').value:=Form19.da
teTimePicker1.Date;Adocommand2.Parameters.ParamByName('Stepen_rodstva').value:=Form19.Combobox1.text;
Adocommand2.Parameters.ParamByName('Adres_rodstvennika').value:=Form19.Edit8.text;
if MessageBox(Handle,'Сохранитьизменения?','Редактирование...',MB_YESNO)=idYES then
begin
Adocommand2.execute;
AdoDataSet1.Close;
AdoDataSet1.Open;
procedure TForm17.ToolButton5Click(Sender: TObject);
begin
sort_direction:='ASC';
adodataSet1.Sort:=b+' '+sort_direction;
end;
procedure TForm17.ToolButton6Click(Sender: TObject);
begin
sort_direction:='DESC';
adodataSet1.Sort:=b+' '+sort_direction;
end;
procedure TForm17.ToolButton7Click(Sender: TObject);
begin
adodataset1.Active:=false;
adodataset1.Active:=true;
sort_direction:='ASC';
adodataset1.Sort:='Kod_rodstvennika'+ ' '+sort_direction;
procedure TForm11.DBGrid1TitleClick(Column: TColumn);
begin
A:='Nazv_otdela';
B:='Nazv_otdela';
end;
procedure TForm11.N2Click(Sender: TObject);

```

```

begin
Form11.Close;
end;
procedure TForm11.N4Click(Sender: TObject);
begin
Form14.Show;
end;
procedure TForm11.N5Click(Sender: TObject);
begin
Form8.Show;
end;
procedure TForm11.N6Click(Sender: TObject);
begin
Form1.Show;
end;
procedure TForm11.N7Click(Sender: TObject);
begin
Form6.Show;
end;
procedure TForm11.N8Click(Sender: TObject);
begin
Form17.Show;
end;
procedure TForm11.N9Click(Sender: TObject);
begin
HTML:=TSTRINGLIST.Create;
HTML.CLEAR;
HTML.ADD('<HTML><TITLE>Отделы</TITLE>');
HTML.ADD('<BODY BGCOLOR="WHITE"><P ALIGN=CENTER><FONT
SIZE="5">Отделыорганизации</P>');
HTML.ADD('<BODY BGCOLOR="WHITE"><P ALIGN=CENTER><FONT
SIZE="4">"Центрагиеныэпидемиологии" БелгородскойобластиНовооскольскогорайона</P>');
HTML.ADD('<TABLE WIDTH=100% BORDER"2" ALIGN="CENTER" BORDER COLOR="BLACK">');
HTML.ADD('<TR>');
HTML.ADD('<TD WIDTH=15% ALIGN=CENTER><FONT SIZE="4"><B>Кодотдела</b></td>');
HTML.ADD('<TD WIDTH=15% ALIGN=CENTER><FONT SIZE="4"><B>Названиеотдела</b></td>');
ADODATASET1.First;
for I:=1 TO ADODATASET1.RECORDCOUNT DO
BEGIN
HTML.ADD('<TR>');
HTML.ADD('<TD WIDTH=25% ALIGN=CENTER>'+ADODDataSet1.Kod_otdela.ASSTRING+'</TD>');
HTML.ADD('<TD WIDTH=25% ALIGN=CENTER>'+ADODDataSet1.Nazv_otdela.ASSTRING+'</TD>');
HTML.ADD('</TR>');
ADODATASET1.NEXT;
END;
HTML.ADD('</TABLE>');
html.Add('<tr>');
html.Add('<td width="410" scope="col"><FONT SIZE=4><div align="left"><span
class="стиль2">Начальник</span></div></FONT></td>');
html.Add('<th width="362" scope="col">&nbsp;&nbsp;&nbsp;</th>');
html.Add('</tr>');
html.Add('<tr>');
html.Add('<td scope="col"><FONT SIZE=4><div align="left"
class="стиль2">отделакадров:</div></FONT></td>');
html.Add('<td width="362" scope="col"><FONT SIZE=4><div align="right"><span
class="стиль2">/ _____ / _____</span></div></FONT SIZE=4></td>');
html.Add('</tr>');
html.Add('</table>');
HTML.ADD('</BODY></HTML>');
FORM20.Caption:='Отделы';
filename:= 'Отделы';
html.savetofile(filename+'.htm');
getdir(0,path);

```

```

form20.Webbrowser1.Navigate(path+'\'+filename+'.htm');
form20.Show;
end;
procedure TForm11.ToolButton1Click(Sender: TObject);
begin
if Form12.showModal=mrOk then
begin
Adocommand1.Parameters.ParamByName('Kod_otdela').value:=Form12.Edit1.text;Adocommand1.Parameters.
ParamByName('Nazv_otdela').value:=Form12.Edit2.text;
Adocommand1.Execute;
Adodataset1.Close;
Adodataset1.Open;
end;
end;
procedure TForm11.ToolButton2Click(Sender: TObject);
begin
Form13.Edit1.text:=ADODataSet1Kod_otdela.AsString;
Form13.Edit2.text:=ADODataSet1Nazv_otdela.AsString;
Begin
if Form13.ShowModal=mrok then
Begin Adocommand2.Parameters.ParamByName('Kod_otdela').value:=Form13.Edit1.text;
Adocommand2.Parameters.ParamByName('Nazv_otdela').value:=Form13.Edit2.text;
if MessageBox(Handle,'Сохранитьизменения?', 'Редактирование...',MB_YESNO)=idYES then
begin
Adocommand2.execute;
AdoDataSet1.Close;
AdoDataSet1.Open;
end;
End;
End;
end
procedure TForm11.ToolButton3Click(Sender: TObject);
begin
if MessageBox(Handle,'Удалитьзапись?', 'Удаление...',MB_YESNO)=IDYES then
beginAdocommand3.Parameters.ParamByName('Kod_otdela').value:=ADODataSet1Kod_otdela.AsString;
Adocommand3.Execute;
AdodataSet1.Close;
Adodataset1.Open;
end;
procedure TForm11.ToolButton5Click(Sender: TObject);
begin
sort_direction:='ASC';
adodataSet1.Sort:=b+' '+sort_direction;
procedure TForm11.ToolButton6Click(Sender: TObject);
begin
sort_direction:='DESC';
adodataSet1.Sort:=b+' '+sort_direction;
procedure TForm11.ToolButton7Click(Sender: TObject);
begin
adodataset1.Active:=false;
adodataset1.Active:=true;
sort_direction:='ASC';
adodataset1.Sort:='Kod_otdela'+ ' '+sort_direction;
end;
procedure TForm6.DBGrid1TitleClick(Column: TColumn);
begin
A:='Nazv_dolgnosti';
B:='Nazv_dolgnosti';
end;
procedure TForm6.N3Click(Sender: TObject);
begin
Form6.Close;
end;

```

```

procedure TForm6.N4Click(Sender: TObject);
begin
Form1.Show;
end;
procedure TForm6.N5Click(Sender: TObject);
begin
Form14.Show;
end;
procedure TForm6.N6Click(Sender: TObject);
begin
Form8.Show;
end;
procedure TForm6.N7Click(Sender: TObject);
begin
Form11.Show;
end;
procedure TForm6.N8Click(Sender: TObject);
begin
Form17.Show;
end;
procedure TForm6.N9Click(Sender: TObject);
begin
HTML:=TSTRINGLIST.Create;
HTML.CLEAR;
HTML.ADD('<P ALIGN=CENTER><FONT
SIZE="5"><B>Списокдолжностейсотрудников</b></FONT></P>');
HTML.ADD('<BODY BGCOLOR="WHITE"><P ALIGN=CENTER><FONT
SIZE="4">"Центрагиgieneиэпидемиологии" БелгородскойобластивНовооскольскомрайоне</P>');
HTML.ADD('<TABLE WIDTH=100% BORDER"2" ALIGN="CENTER" BORDER COLOR="BLACK">');
HTML.ADD('<TR>');

HTML.ADD('<TD WIDTH=15% ALIGN=CENTER><FONT SIZE="4"><B>Названиедолжности</b></td>');
HTML.ADD('<TD WIDTH=15% ALIGN=CENTER><FONT
SIZE="4"><B>Выполняемыефункции</b></td>');
ADODATASET1.First;
for I:=1 TO ADODATASET1.RECORDCOUNT DO
BEGIN
HTML.ADD('<TR>');
HTML.ADD('<TD WIDTH=20% ALIGN=CENTER>'+ADODDataSet1.Nazv_dolgnosti.ASSTRING+'</TD>');
HTML.ADD('<TD WIDTH=20%
ALIGN=CENTER>'+ADODDataSet1.Vipolnyamie_funkcii.ASSTRING+'</TD>');
HTML.ADD('</TR>');
ADODATASET1.NEXT;
END;
HTML.ADD('</TABLE>');
html.Add('<table width=100% border="0">');
html.Add('<tr>');
html.Add('<td width="410" scope="col"><FONT SIZE=4><div align="left"><span
class="стиль2">Начальник</span></div></FONT></td>');
html.Add('<th width="362" scope="col">&nbsp;</th>');
html.Add('</tr>');
html.Add('<tr>');
html.Add('<td scope="col"><FONT SIZE=4><div align="left"
class="стиль2">отделакадров:</div></FONT></td>');
html.Add('<td width="362" scope="col"><FONT SIZE=4><div align="right"><span
class="стиль2">/_____/_____/</span></div></FONT SIZE=4></td>');
html.Add('</tr>');
html.Add('</table>');
HTML.ADD('</BODY></HTML>');
FORM20.Caption:='Должности';
filename:= 'Должности';
html.savetofile(filename+'.htm');
getdir(0,path);

```

```

form20.Webbrowser1.Navigate(path+'\'+filename+'.htm');
form20.Show;
end;
procedure TForm6.ToolButton1Click(Sender: TObject);
begin
if Form9. showModal=mrOk then
begin
Adocommand1.Parameters.ParamByName('Kod_dolgnosti').value:=Form9.Edit1.text;Adocommand1.Parameter
s.ParamByName('Nazv_dolgnosti').value:=Form9.Edit2.text;Adocommand1.Parameters.ParamByName('Vipolnyamie_fun
kcii').value:=Form9.Edit3.text;
AdoCommand1.Execute;
Adodataset1.Close;
procedure TForm6.ToolButton2Click(Sender: TObject);
begin
Form10.Edit1.text:=ADODataSet1Kod_dolgnosti.AsString;
Form10.Edit2.text:=ADODataSet1Nazv_dolgnosti.AsString;
Form10.Edit3.text:=ADODataSet1Vipolnyamie_funkcii.Asstring;
Begin
if Form10.ShowModal=mrok then
BeginAdocommand2.Parameters.ParamByName('Kod_dolgnosti').value:=Form10.Edit1.text;Adocommand2.Par
ameters.ParamByName('Nazv_dolgnosti').value:=Form10.Edit2.text;Adocommand2.Parameters.ParamByName('Vipolnya
mie_funkcii').value:=Form10.Edit3.text;
if MessageBox(Handle,'Сохранитьизменения?','Редактирование...','MB_YESNO)=idYES then
begin
Adocommand2.execute;
AdoDataSet1.Close;
AdoDataSet1.Open;
end;
End;
procedure TForm6.ToolButton3Click(Sender: TObject);
begin
if MessageBox(Handle,'Удалитьзапись?','Удаление...','MB_YESNO)=IDYES then
begin
Adocommand3.Parameters.ParamByName('Kod_dolgnosti').value:=ADODataSet1Kod_dolgnosti.asString;
Adocommand3.Execute;
AdodataSet1.Close;
Adodataset1.Open;
end;
end
procedure TForm6.ToolButton5Click(Sender: TObject);
begin
sort_direction:='ASC';
adodataSet1.Sort:=b+' '+sort_direction;
end;
procedure TForm6.ToolButton6Click(Sender: TObject);
begin
sort_direction:='DESC';
adodataSet1.Sort:=b+' '+sort_direction;
end;
procedure TForm6.ToolButton7Click(Sender: TObject);
begin
adodataset1.Active:=false;
adodataset1.Active:=true;
sort_direction:='ASC';
adodataset1.Sort:='Kod_dolgnosti'+ ' '+sort_direction;
end;
procedure TForm14.DBGrid1TitleClick(Column: TColumn);
begin
A:='Naim_oborudovaniya';
B:='Naim_oborudovaniya';
end;
procedure TForm14.Jnxtm1Click(Sender: TObject);
begin

```

```

HTML:=TSTRINGLIST.Create;
HTML.CLEAR;
HTML.ADD('<HTML><TITLE>Оборудование</TITLE>');
HTML.ADD('<BODY BGCOLOR="WHITE"><P ALIGN=CENTER><FONT SIZE="5">Оборудование,
относящиеся к сотруднику</P>');
HTML.ADD('<BODY BGCOLOR="WHITE"><P ALIGN=CENTER><FONT
SIZE="4">"Центра гигиены и эпидемиологии" Белгородской области Новооскольского района</P>');
HTML.ADD('<TABLE WIDTH=100% BORDER="2" ALIGN="CENTER" BORDER COLOR="BLACK">');
HTML.ADD('<TR>');
HTML.ADD('<TD WIDTH=15% ALIGN=CENTER><FONT
SIZE="4"><B>Наименование оборудования</b></td>');
HTML.ADD('<TD WIDTH=15% ALIGN=CENTER><FONT SIZE="4"><B>Изготовитель</b></td>');
HTML.ADD('<TD WIDTH=15% ALIGN=CENTER><FONT SIZE="4"><B>Дата эксплуатации</b></td>');
HTML.ADD('<TD WIDTH=15% ALIGN=CENTER><FONT
SIZE="4"><B>Технические характеристики</b></td>');
HTML.ADD('<TD WIDTH=15% ALIGN=CENTER><FONT SIZE="4"><B>Документация</b></td>');
HTML.ADD('<TD WIDTH=15% ALIGN=CENTER><FONT
SIZE="4"><B>Фамилия сотрудника</b></td>');
ADODATASET1.First;
for I:=1 TO ADODATASET1.RECORDCOUNT DO
BEGIN
HTML.ADD('<TR>');
HTML.ADD('<TD WIDTH=20%
ALIGN=CENTER>'+ADODataset1Data_ekspluatcii.ASSTRING+'</TD>');
HTML.ADD('<TD WIDTH=20% ALIGN=CENTER>'+ADODataset1Teh_harakteristiki.ASSTRING+'</TD>');
HTML.ADD('<TD WIDTH=20% ALIGN=CENTER>'+ADODataset1Dokumentaciya.ASSTRING+'</TD>');
HTML.ADD('<TD WIDTH=20% ALIGN=CENTER>'+ADODataset1Familiya.ASSTRING+'</TD>');
HTML.ADD('</TR>');
ADODATASET1.NEXT;
HTML.ADD('</TABLE>');
html.Add('<table width=100% border="0">');
html.Add('<tr>');
html.Add('<td width="410" scope="col"><FONT SIZE=4><div align="left"><span
class="стиль2">Начальник</span></div></FONT></td>');
html.Add('<th width="362" scope="col">&nbsp;&nbsp;&nbsp;</th>');
html.Add('<tr>');
html.Add('<tr>');
html.Add('<td scope="col"><FONT SIZE=4><div align="left"
class="стиль2">отдела кадров:</div></FONT></td>');
html.Add('<td width="362" scope="col"><FONT SIZE=4><div align="right"><span
class="стиль2">/_____/_____/</span></div><FONT SIZE=4></td>');
html.Add('</tr>');
html.Add('</table>');
HTML.ADD('</BODY></HTML>');
FORM20.Caption:='Оборудование';
filename:='Оборудование';
html.savetofile(filename+'.htm');
getdir(0,path);
form20.Webbrowser1.Navigate(path+'\'+filename+'.htm');
form20.Show;
end;
procedure TForm14.N2Click(Sender: TObject);
begin
Form14.Close;
end;
procedure TForm14.N4Click(Sender: TObject);
begin
Form8.Show;
end;
procedure TForm14.N5Click(Sender: TObject);
begin
Form1.Show;
end;
procedure TForm14.N6Click(Sender: TObject);

```



```

begin
Form11.Show;
procedure TForm14.N7Click(Sender: TObject);
begin
Form6.Show;
end;
procedure TForm14.N8Click(Sender: TObject);
begin
Form17.Show;
end;
procedure TForm14.ToolButton1Click(Sender: TObject);
begin
if Form15. showModal=mrOk then
beginAdocommand1.Parameters.ParamByName('Kod_oborudovaniya').value:=Form15.Edit1.text;Adocommand
1.Parameters.ParamByName('Naim_oborudovaniya').value:=Form15.Edit2.text;Adocommand1.Parameters.ParamByName
('Izgotovitel').value:=Form15.Edit3.textAdocommand1.Parameters.ParamByName('Data_ekscpluatacii').value:=Form15.Ed
it4.Text;Adocommand1.Parameters.ParamByName('Teh_harakteristiki').value:=Form15.Edit5.text;Adocommand1.Paramet
ers.ParamByName('Dokumentaciya').value:=Form15.Edit6.text;Adocommand1.Parameters.ParamByName('Kod_sotrudnik
a').value:=Form21.ADODataSet1Kod_sotrudnika.asstring;
AdoCommand1.Execute;
Adodataset1.Close;
Adodataset1.Open;
procedure TForm14.ToolButton2Click(Sender: TObject);
begin
Form16.Edit1.text:=ADODataset1Kod_oborudovaniya.AsString;
Form16.Edit2.text:=ADODataset1Naim_oborudovaniya.AsString;
Form16.Edit3.Text:=ADODataset1Izgotovitel.AsString;
Form16.Edit4.text:=ADODataset1Data_ekscpluatacii.Asstring;
Form16.Edit5.text:=ADODataset1Teh_harakteristiki.AsString;
Form16.Edit6.text:=ADODataset1Dokumentaciya.AsString;
Form16.Edit7.text:=ADODataset1Kod_sotrudnika.AsString;
Begin
if Form16.ShowModal=mrok then
Begin
Adocommand2.Parameters.ParamByName('Kod_oborudovaniya').value:=Form16.Edit1.text;Adocommand2.Parameters.Pa
ramByName('Naim_oborudovaniya').value:=Form16.Edit2.text;Adocommand2.Parameters.ParamByName('Izgotovitel').va
lue:=Form16.Edit3.textAdocommand2.Parameters.ParamByName('Data_ekscpluatacii').value:=Form16.Edit4.text;Adocom
mand2.Parameters.ParamByName('Teh_harakteristiki').value:=Form16.Edit5.text;Adocommand2.Parameters.ParamByNa
me('Dokumentaciya').value:=Form16.Edit6.text;Adocommand2.Parameters.ParamByName('Kod_sotrudnika').value:=Form
21.ADODataSet1Kod_sotrudnika.asstring;
if MessageBox(Handle,'Сохранитьизменения?','Редактирование...','MB_YESNO)=idYES then
begin
begin
Adocommand2.execute;
AdoDataSet1.Close;
AdoDataSet1.Open;
end;
End;
procedure TForm14.ToolButton3Click(Sender: TObject);
begin
sort_direction:='ASC';
adodataSet1.Sort:='b+' +sort_direction;
end;
procedure TForm14.ToolButton4Click(Sender: TObject);
begin
sort_direction:='DESC';
adodataSet1.Sort:='b+' +sort_direction;
procedure TForm14.ToolButton5Click(Sender: TObject);
begin
Adocommand3.Parameters.ParamByName('Kod_oborudovaniya').value:=ADODataset1Kod_oborudovaniya.as
String;
Adocommand3.Execute;
AdodataSet1.Close; Adodataset1.Open;
end;

```

```
procedure TForm14.ToolButton6Click(Sender: TObject);
begin
adodataset1.Active:=false;
adodataset1.Active:=true;
sort_direction:='ASC';
adodataset1.Sort:='Kod_oborudovaniya'+'+'+sort_direction;
procedure TForm14.ToolButton8Click(Sender: TObject);
begin
Form37.ADODataSet1.Close;
Form37.ADODataSet1.Parameters.ParamByName('Kod_oborudovaniya').Value:=ADODataSet1Kod_oborudova
niya.AsString;
Form37.ADODataSet1.Open;
Form37.Show;
end
```

Приложение Ж

Методы испытаний системы

Таблица Ж.1 - Проверка функции запуска приложения

Входные данные	Выходные данные
Запуск приложения SES.exe	Отображение главного окна программы.

Таблица Ж.2 - Проверка подтверждения подлинности пользователя

Входные данные	Выходные данные
В главном окне программы нажатие кнопки «Открыть АИС».	Отображение формы для ввода логина и пароля
Ввод неверного логина «Школа». Пароль 546681	Отображение сообщения: «Неправильный пароль! Повторите ввод!».
Ввод верного логина «Кудрявцева» и пароля 1001	Отображение главной формы с данными о сотрудниках предприятия

Таблица Ж.3 - Проверка функции ввода нового сотрудника

Входные данные	Выходные данные
Вызов формы «Сотрудники».	Отображение формы «Сотрудники»
Нажатие кнопки «Новый Сотрудник»	Открытие формы «Добавление записи нового сотрудника»
Ввод данных о новом владельце: Фамилия - Иванов, имя – Алексей, отчество – Александрович, дата рождения – 16.03.72, специальность- химик-эксперт, Семейное положение – женат, возраст - 40, пол – мужской, национальность- русский, адрес проживания - народная 50, название должности – химик эксперт, название отдела - паразитологический отдел, Вид образования – высшее,	Отображение в таблице «Sotrudniki» записи: Фамилия - Иванов, имя – Алексей, отчество – Александрович, дата рождения – 16.03.72, специальность- химик-эксперт, Семейное положение – женат, возраст - 40, пол – мужской, национальность- русский, адрес проживания - народная 50, название должности – химик эксперт, название отдела - паразитологический отдел, Вид образования – высшее,

Таблица Ж.4 - Проверка функции редактирования информации о
сотруднике

Входные данные	Выходные данные
1	2
Вызов формы «Редактирование сотрудника»	Отображение формы «Редактирование записи сотрудника»
Редактирование данных о текущем владельце (изменение специальности): Фамилия - Иванов, имя – Алексей, отчество – Александрович, дата рождения – 16.03.72, специальность- врач-бактериолог, Семейное положение – женат, возраст - 40, пол – мужской, национальность- русский, адрес проживания - народная 50, название должности – химик эксперт, название отдела - паразитологический отдел, Вид образования – высшее,.	
Нажатие на кнопку «Ок».	Отображение сообщения «Сохранить изменения?».
Нажатие на кнопку «Да».	Отображение в таблице «Sotrudniki» обновленной записи о сотруднике:): Фамилия - Иванов, имя – Алексей, отчество – Александрович, дата рождения – 16.03.72, специальность- врач- бактериолог, Семейное положение – женат, возраст - 40, пол – мужской, национальность- русский, адрес проживания - народная 50, название должности – химик эксперт, название отдела - паразитологический отдел, Вид образования – высшее,.

Таблица Ж.4 - Продолжение

1	2
Нажатие на кнопку «Нет».	Отображение в таблице «Sotrudniki» текущей записи без обновлений овладельце: Фамилия - Иванов, имя – Алексей, отчество – Александрович, дата рождения – 16.03.72, специальность- химик-эксперт, Семейное положение – женат, возраст - 40, пол – мужской, национальность- русский, адрес проживания - народная 50, название должности – химик эксперт, название отдела - паразитологический отдел, Вид образования – высшее

Таблица Ж.5 - Проверка функции удаления информации о сотруднике

Входные данные	Выходные данные
Форма «Сотрудники» выбор владельца, Иванова Алексея Александровича, для удаления нажатие кнопки «Удалить сотрудника»	Отображение сообщения «Удалить запись?».
Отмена нажатием на кнопку «Нет»	В таблице «Sotrudniki» остались данные о текущем сотруднике.
Подтверждение нажатием на кнопку «Да»	Закрытие диалогового окна удаления сотрудника
Данные формы «Сотрудники»	В таблице «Sotrudniki» больше нет данных о сотруднике иванове Алексее Александровиче.

Таблица Ж.6 - Проверка функции фильтрация владельцев по дате рождения

Входные данные	Выходные данные
Форма «Сотрудники» нажатие кнопки «Фильтрация по дате рождения»	Отображение диалогового окна с выбором параметров для фильтрации
Подтверждение фильтрации по выбранному диапазону от 11.02.1968 до 11.02.1969 нажатием на кнопку «Ок»	Закрытие диалогового окна фильтрации по дате рождения
Данные формы «Сотрудники»	В таблице «Sotrudniki» остались данные, соответствующие параметрам фильтрации.

Таблица Ж.7 - Проверка функции поиска сотрудника

Входные данные	Выходные данные
Форма «Сотрудники» ввод Фамилия - Темботова строку поиска	В таблице «Sotrudniki» остались только данные, соответствующие работнику Темботовой

Таблица Ж.8 - Проверка функции сортировки фамилии сотрудников по
возрастанию

Входные данные	Выходные данные
Форма «Сотрудники» нажатие кнопки «Сортировать по возрастанию»	В таблице «Sotrudniki» данные отображаются согласно сортировки фамилий по возрастанию.

Таблица Ж.9 - Проверка функции формирования отчета личного листка
владельца

Входные данные	Выходные данные
Вызов формы «Личный листок сотрудника»	Отображение формы «Личный листок сотрудника». В поле вывода отчета шаблон заполнен относительно выбранного сотрудника.