Федеральное государственное автономное
образовательное учреждение
высшего образования
«СИБИРСКИЙ ФЕДЕРАЛЬНЫЙ УНИВЕРСИТЕТ»

Институт Космических и Информационных Технологий
институт
Информационные Системы
кафедра

УТВЕРЖДАЮ
Зав. кафедрой ИС
_____ Виденин С. А.
подпись инициалы, фамилия
« ____» _____ 2016 г.

ДИПЛОМНЫЙ ПРОЕКТ

230201.65 Информационные системы и технологии

Разработка информационной системы учета заявок на техническую поддержку для сервисного центра

Пояснительная записка

Руководитель _____ А. А. Латынцев
подпись, дата инициалы, фамилия

Выпускник _____ Е. В. Варнавская
подпись, дата инициалы, фамилия

Нормоконтролер _____ Ю. В. Шмагрис
подпись, дата инициалы, фамилия

Красноярск 2016

Федеральное государственное автономное
образовательное учреждение
высшего образования
«СИБИРСКИЙ ФЕДЕРАЛЬНЫЙ УНИВЕРСИТЕТ»

Институт космических и информационных технологий

Кафедра «Информационные системы»

УТВЕРЖДАЮ
Заведующий кафедрой ИС
_____ С. А. Виденин
подпись
«5» декабря 2015 г.

ЗАДАНИЕ

НА ВЫПУСКНУЮ КВАЛИФИКАЦИОННУЮ РАБОТУ

в форме дипломного проекта

Студенту (ке) Варнавской Елене Владимировне

Группа: ЗКИ10-02-1 Специальность: 230201.65 «Информационные системы и технологии»

Тема выпускной квалификационной работы: «Разработка информационной системы учета заявок на техническую поддержку для сервисного центра»

Утверждена приказом по университету №4043/с от 24 марта 2016г.

Руководитель ВКР: А.А. Латынцев, к. т. н., доцент НУЛ «Систем искусственного интеллекта» кафедры «Системы искусственного интеллекта».

Исходные данные для ВКР: список требований к разрабатываемой системе, методические указания научного руководителя.

Перечень разделов ВКР: введение, аналитическая часть, проектирование и реализация АС, среда разработки PHP, серверная часть, руководство пользователя, заключение, список использованной литературы.

Перечень графического или иллюстрированного материала с указанием основных чертежей, плакатов, слайдов: презентация, выполненная в Microsoft Office PowerPoint 2007.

Руководитель ВКР _____ А.А. Латынцев
(подпись)

Задание принял к исполнению _____ Е. В. Варнавская
(подпись)

« ___ » _____ 20__ г.

РЕФЕРАТ

Выпускная квалификационная работа по теме «Разработка информационной системы учета заявок на техническую поддержку для сервисного центра» содержит 47 страниц текстового документа, 0 приложений, 16 использованных источников, 0 листов графического материала.

Титульный слайд, актуальность, цели и задачи, обзор аналогичного по, логическая схема бд, выбор средств разработки, uml-диаграмма, окно приветствия, окно списка заявок, окно редактирования заявки, справочник «подразделение», справочник «операции», окно расширенного поиска, слайд заключение.

Цель проекта – разработать информационную систему учета заявок на техническую поддержку для сервисного центра.

Основные задачи:

1. Собрать информацию для решения поставленной цели;
2. Разработать базу данных;
3. Разработать программный продукт;
4. Проверка продукта на наличие ошибок.

Основные результаты:

1. Создан программный продукт для сервисного центра;
2. Программный продукт внедрен для использования в организации.

СОДЕРЖАНИЕ

ВВЕДЕНИЕ 4
1 Аналитическая часть 5
1.1 Характеристика объекта исследования 5
1.2 Определение цели и задач проектирования ИС 5
1.3 Обзор существующего программного обеспечения 7
1.4 Выбор программных средств реализации АС 8
1.5 Техническое задание на создание АИС 10
1.5.1 Общие положения 10
1.5.2 Назначение и цели создания системы 12
1.5.3 Характеристика объекта автоматизации 12
1.5.4 Требования к системе 13
1.5.5 Порядок контроля и приемки системы 18
1.5.6 Требования к документированию 18
1.6 Выводы по главе 18
2 Проектирование и реализация информационной системы 19
2.1 Логическое проектирование базы данных 20
2.2 Физическое проектирование базы данных 24
2.3 Серверная часть 30
2.3.1 Установка *Oracle Database Express Edition* 31
2.3.2 Установка переменных окружения для *Oracle Database Express Edition* 33
2.3.3 Удаленный доступ к Oracle Database Express Edition 34
2.4 Руководство пользователя 36
2.5 Тестирование программного продукта 43
ЗАКЛЮЧЕНИЕ 45
СПИСОК ИСПОЛЬЗОВАННЫХ ИСТОЧНИКОВ 46

ВВЕДЕНИЕ

В настоящее время компьютерные технологии применяются в жизни все чаще и чаще. Программное обеспечение используется для решения разнообразных задач почти во всех отраслях производства, экономики, образования и т.д. Это значительно упрощает и автоматизирует ранее громоздкую работу, которую приходилось делать вручную. Развитие компьютерных технологий продолжается со скоростью прогресса все быстрее и быстрее.

Результатом использования средств вычислительной техники во многих самых различных сферах деятельности человека является значительное снижение затрат труда на выполняемые операции, ускорение процессов получения необходимых результатов, а также сведение к минимуму возможности возникновения каких-либо неточностей, иногда возникающих при работе человека из-за невнимательности.

На сегодняшний момент существует огромное множество программных продуктов для автоматизации деятельности предприятий. Они очень полезны, так как автоматизируют работу на всех этапах функционирования предприятия.

Цель дипломного проекта состоит в автоматизации деятельности службы технической поддержки.

Для достижения поставленной цели необходимо решить следующие задачи:

- исследовать предметную область;
- произвести обзор существующего программного обеспечения, подходящего для решения поставленной задачи автоматизации;
- выбрать инструментальные средства разработки программного продукта;
- выполнить проектирование базы данных;
- разработать и протестировать программное обеспечение.

1 Аналитическая часть

1.1 Характеристика объекта исследования

С целью обеспечения обслуживания всего оборудования, на большинстве предприятии существует сектор технического обслуживания компьютерной техники.

Отдел технического обслуживания компьютерной техники выполняет следующие функции:

- ведет прием заявок на обслуживание;
- ведет учет компьютерной техники;
- обслуживает компьютерную технику (закупка, ремонт, замена).

Основой для учета, контроля и планирования служат всевозможные картотеки, регистрационные журналы, списки и т.д., которые постепенно накапливаются и обновляются. При большом объеме информации процессы поиска и обобщения необходимых сведений, осуществляемые вручную, представляют собой довольно трудоемкий процесс. Кроме того, при обработке информации вручную, существует вероятность потери некоторых сведений, подчас играющих важную роль. Во избежание подобных ситуаций и в целях экономии времени, необходимо автоматизировать деятельность отдела технического обслуживания компьютерной техники.

1.2 Определение цели и задач проектирования ИС

В соответствии с поставленной целью в рамках дипломного проекта требуется решить следующие задачи:

1. Исследовать предметную область.
2. Произвести обзор существующего программного обеспечения, подходящего для решения поставленной задачи автоматизации.

3. Выбрать инструментальные средства разработки программного продукта.

4. Выполнить проектирование базы данных.

5. Разработать и протестировать программное обеспечение.

Поскольку основным результатом дипломного проекта должна явиться информационная система учета заявок на обслуживание, службы технической поддержки предприятия, необходимо сформулировать основные требования, предъявляемые к функциональным возможностям разрабатываемого программного обеспечения.

В первую очередь следует отметить, что подлежащая разработке программная система характеризуется тем, что она находится на сервере и обеспечивает работу администратора и многих клиентов. Администратор может, в частности, определять право доступа для каждого пользователя или для группы пользователей, пополнять данные базы, следить за безопасностью информации. Клиентами базы данных являются, прежде всего, работники службы технического обслуживания компьютерной техники, а также пользователи этой техники. Они работают на персональных компьютерах и могут общаться с базой данных через *WEB*-интерфейс, с использованием своего учетного имени и пароля. Каждый из пользователей имеет определенные права доступа к базе данных, которые определяет администратор.

База данных должна предоставлять пользователю – работнику службы технического обслуживания компьютерной техники – исчерпывающую информацию, необходимую для быстрого осуществления работ по обслуживанию и ремонту техники, а также информацию по технике в организации. Пользователь-«клиент» службы технического обслуживания компьютерной техники должен получить возможность быстро подать заявку на обслуживание с уверенностью, что она не останется без внимания.

Другой отличительной чертой разрабатываемой системы является то, что она централизованная, то есть при работе в системе пользователи

общаются с базой данных через *WEB*-интерфейс и все операции добавления, удаления, обновления и поиска информации в базе данных происходят на сервере, где находится база данных. Перечисленные операции должны выполняться достаточно быстро, потому что многие пользователи работают одновременно с базой данных.

Еще одна отличительная черта разрабатываемой системы от существующих систем учета компьютерной техники заключается в том, что каждая единица комплектующих является самостоятельной единицей учета.

1.3 Обзор существующего программного обеспечения

На сегодняшний день на рынке программного обеспечения предлагается целый спектр разнообразных качественных программных продуктов, позволяющих автоматизировать ведение различного вида учета на предприятиях и в организациях.

Ориентируясь на специфику предметной области, рассматриваемой в данной работе, остановимся на двух программных продуктах, которые могут быть использованы в работе отдела по обслуживанию компьютерной техники:

- Система «*HTP*-Сервис» предназначена для автоматизации приема, учета и обслуживания обращений авторизованных пользователей по вопросам эксплуатации оборудования, для хранения и оперативного представления информации о составе оборудования пользователей, договорах обслуживания, а также для планирования работы службы поддержки;
- *LanDocs* – это система, предназначенная для управления данными и документами, возникающими в процессах технической поддержки. Использование данной системы позволяет упорядочить и документировать процессы технической поддержки, результатом чего, в свою очередь,

является существенное повышение качества выполнения проектных и сервисных работ.

В качестве основных недостатков готового программного обеспечения, предлагаемого на рынке программных продуктов, можно назвать такие как:

- высокие цены на покупку, установку и сопровождение;
- необходимость настройки под характерные особенности предприятия. При этом данные услуги подчас дороже, чем сам программный продукт;
- слишком широкий спектр реализуемых функций, которые большинство предприятий никогда не будут использовать, переплачивая за них серьезные деньги

1.4 Выбор программных средств реализации АС

Выбор СУБД является одним из важнейших моментов в разработке проекта, так как он принципиальным образом влияет на весь процесс проектирования и реализации системы. Теоретически при выборе СУБД нужно принимать во внимание десятки факторов. Но практически разработчики руководствуются лишь собственной интуицией и несколькими наиболее важными критериями, к которым, в частности, относятся:

- тип модели данных, которую поддерживает данная СУБД, её соответствие потребностям рассматриваемой предметной области;
- характеристики производительности системы;
- запас функциональных возможностей для дальнейшего развития ИС;
- степень оснащённости системы инструментарием для персонала администрирования данными;
- удобство и надежность СУБД в эксплуатации;
- стоимость СУБД и дополнительного программного обеспечения.

Анализ информационных задач показал, что для реализации требуемых функций подходят СУБД технологии клиент-сервер, такие как *Microsoft SQL Server, Oracle, IBM DB2, Sybase* и др. Все они поддерживают клиент-

серверную модель данных и предоставляют разнообразные возможности для работы с данными. Выбор остановился на *Oracle Database XE* 10*G*.

При выборе среды разработки программного продукта были учтены следующие факторы:

1. Сложность и трудоемкость процесса проектирования программного обеспечения для конкретной среды.
2. Наличие инструментальных средств разработки программного обеспечения.
3. Возможность внесения корректив в программу в процессе эксплуатации.
4. Наличие средств проектирования пользовательского интерфейса.
5. Скорость выполнения программы.
6. Надежность работы программы и защищенность от программных сбоев.
7. Возможность масштабирования приложения без глобального обновления клиентских ПК.
8. Гибкость проектирования интерфейса.

В качестве среды функционирования разрабатываемого программного продукта была выбрана ОС *Windows*, поскольку данная система является наиболее распространенной и уже имеется на ПЭВМ, с помощью которой разрабатывалась программа. Также данная среда имеет большое количество систем разработки *web*-приложений. Разработка в среде *Windows* значительно облегчается наличием систем визуального проектирования.

На сегодняшний момент имеется несколько развитых языков программирования, позволяющих создавать полноценные программы, предназначенные для работы в среде *Windows*. *PHP* - язык программирования, используемый на стороне *WEB*-сервера для динамической генерации *HTML*-страниц. Об этом говорит и расшифровка его названия: *PHP - Personal HyperText Processor*

PHP - один из немногих языков программирования, созданных специально для разработки веб-приложений. Поэтому он включает в себя все функции, необходимые именно для работы на веб-сервере, и при этом лишен избыточности, свойственной многим его конкурентам.

Особенность *PHP* - то, что его команды включаются в обычные *HTML*-страницы с помощью специальных тегов, которые и заставляют *PHP*-машину выполнять на сервере нужные действия. Программам на *PHP* не нужны специальные *CGI*-директории с особыми правами доступа. Более того, на одной страничке можно произвольно чередовать "простой" *HTML* и *PHP*-код.

PHP не зависит от платформы. *PHP* прекрасно интегрируется во все популярные веб-серверы: *Apache* и *IIS*, *Zens* и *Netscape Enterprise Server*, работает под *Windows* и *OS*/2, *MacOS* и практически всеми *UNIX*-подобными системами. Как следствие - *PHP* работает практически у всех хост-провайдеров, разрешающих собственные выполняемые скрипты.

PHP интегрируется практически со всеми современными интернет-технологиями. *PHP* поддерживает большинство современных веб-протоколов: *IMAP*, *FTP*, *POP*, *XML*, *SNMP* и другие. *PHP* прекрасно работает с базами данных. Широкий спектр СУБД, поддержка которых был реализован в *PHP*, делает язык универсальным средством для создания веб-приложений. Всемирным сообществом, *PHP* признан безусловным лидером среди языков веб-программирования.

1.5 Техническое задание на создание АИС

1.5.1 Общие положения

Полное наименование системы и ее условное обозначение

Полное наименование системы: «Разработка информационной системы учета заявок на техническую поддержку для сервисного центра».

Перечень документов, на основании которых создается система

Основанием для разработки информационной системы является задание на дипломный проект.

Порядок оформления и предъявления заказчику результатов работ по созданию системы

Система передается в виде функционирующего комплекса на базе средств вычислительной техники Заказчика и Исполнителя в сроки, установленные заданием на дипломный проект. Приемка системы осуществляется комиссией в составе уполномоченных представителей Заказчика и Исполнителя. Порядок предъявления системы, ее испытаний и окончательной приемки определен в настоящем ТЗ. Совместно с предъявлением системы производится сдача разработанного Исполнителем комплекта документации согласно определениям настоящего ТЗ.

Перечень нормативно-технических документов, методических материалов, использованных при разработке ТЗ

При разработке автоматизированной системы и создании проектно-эксплуатационной документации Исполнитель должен руководствоваться требованиями следующих нормативных документов:

- ГОСТ 19.201–78. Техническое задание. Требования к содержанию и оформлению;
- ГОСТ 34.601–90. Комплекс стандартов на автоматизированные системы. Автоматизированные системы. Стадии создания;
- ГОСТ 34.201–89. Информационная технология. Комплекс стандартов на автоматизированные системы. Виды, комплексность и обозначение документов при создании автоматизированных систем;
- РД 50–34.698–90. Методические указания. Информационная технология. Комплекс стандартов на автоматизированные системы. Автоматизированные системы. Требования к содержанию документов.

Определения, обозначения и сокращения

В таблице 1 представлен список используемых сокращений.

Таблица 1 – Список используемых сокращений

№	Сокращение	Расшифровка
1	ТЗ	Техническое задание
2	ИС	Информационная система
3	ВКР	Выпускная квалификационная работа
4	ОС	Операционная система
5	НСД	Несанкционированный доступ
6	ER	Инфологическое моделирование
7	УЗ	Учетная запись
8	ПК	Персональный компьютер
9	ПКЗИ	Персональный ключ защиты информации
10	ПО	Программное обеспечение
11	УКТ	Учет компьютерной техники
12	СЦ	Сервисный центр

1.5.2 Назначение и цели создания системы

Назначение системы

Конфигурация разрабатывается для упрощения и ускорения следующих процессов:

- регистрация заявок поступающих в СЦ;
- назначение исполнителя заявки;
- автоматическое оповещение заказчика о всех изменениях в заявке;
- обеспечение учета техники;
- формирование отчетов в различных разрезах.

Информационная система представляет собой программное решение, предназначенное для автоматизации учета заявок поступающих в сервисный центр.

1.5.3 Характеристика объекта автоматизации

Объектом автоматизации является процесс учета заявок, в ходе которого происходит выполнение действий, связанных с внесением необходимой для выполнения заявки информации в базу данных.

Все процессы осуществляются оператор и исполнителем заявки.

1.5.4 Требования к системе

Перечень подсистем, их назначение и основные характеристики

В состав разрабатываемой системы должны входить следующие подсистемы:

- подсистема хранения данных;
- подсистема учета заявок;
- подсистема отчетности.

Подсистема хранения данных предназначена для хранения сведений об исполнителях, заказчиках и технике, с которой производились какие-либо работы. Подсистема учета заявок предназначена для хранения информации о текущих и выполненных работах.

Подсистема формирования отчетности предназначена для создания и формирования отчетов в виде удобном для вывода на печатающие устройства на основе данных информационной системы.

Требования по диагностированию системы

Информационная система должна предоставлять инструменты диагностирования основных процессов системы.

Перспективы развития, модернизации системы

Информационная система должна реализовывать возможность дальнейшей модернизации, как программного обеспечения, так и комплекса технических средств. Также необходимо предусмотреть возможность увеличения производительности системы путем её масштабирования.

Требования к численности и квалификации персонала системы

Численность и квалификация персонала системы должны определяться с учетом следующих требований:

- структура и конфигурация системы должны быть спроектированы и реализованы с целью минимизации количественного состава обслуживающего персонала;

- для администрирования системы к администратору должны предъявляться требования по знанию всех особенностей функционирования элементов, входящих в состав администрируемых компонентов системы;
- аппаратно-программный комплекс системы не должен требовать круглосуточного обслуживания и присутствия администраторов.

Штатный состав персонала, эксплуатирующего систему, должен формироваться на основании нормативных документов Российской Федерации и Трудового кодекса.

Система реализуется на серверном оборудовании, поэтому требования к организации труда и режима отдыха при работе с ней должны устанавливаться, исходя из требований к организации труда и режима отдыха при работе с этим типом средств вычислительной техники.

Пользователи системы должны иметь опыт работы с персональным компьютером на базе операционных систем Microsoft Windows на уровне квалифицированного пользователя и свободно осуществлять базовые операции.

Рекомендуемая численность для эксплуатации:

- администратор – 1 человек;
- пользователь – число человек не ограничено.

Показатели назначения

Система должна предусматривать возможность масштабирования по производительности и объему обрабатываемой информации без модификации ее программного обеспечения путем модернизации используемого комплекса технических средств. Возможности масштабирования должны обеспечиваться средствами используемого базового программного обеспечения.

Требования к надежности

Система должна сохранять работоспособность и обеспечивать восстановление своих функций при возникновении следующих внештатных ситуаций:

- при сбоях в системе электроснабжения аппаратной части, приводящих к перезагрузке ОС, восстановление программы должно происходить после перезапуска ОС и запуска исполняемого файла системы;
- при ошибках в работе аппаратных средств (кроме носителей, данных и программ) восстановление функции системы возлагается на ОС;
- при ошибках, связанных с программным обеспечением (ОС и драйверы устройств), восстановление работоспособности возлагается на ОС.

Для защиты аппаратуры от бросков напряжения и коммутационных помех должны применяться сетевые фильтры.

Требования к безопасности

Все внешние элементы технических средств системы, находящиеся под напряжением, должны иметь защиту от случайного прикосновения, а сами технические средства иметь защитное заземление в соответствии с ГОСТ 12.1.030–81. Система электропитания должна обеспечивать защитное отключение при перегрузках и коротких замыканиях в цепях нагрузки, а также аварийное ручное отключение.

Требования к эксплуатации, техническому обслуживанию

Техническая и физическая защита аппаратных компонентов системы, носителей данных, бесперебойное энергоснабжение, резервирование ресурсов, текущее обслуживание реализуется техническими и организационными средствами, предусмотренными в инфраструктуре Заказчика.

Периодическое техническое обслуживание и тестирование технических средств должны включать в себя обслуживание и тестирование всех используемых средств, включая рабочие станции, серверы, кабельные системы и сетевое оборудование.

Размещение помещений и их оборудование должны исключать возможность бесконтрольного проникновения в них посторонних. Размещение оборудования, технических средств должно соответствовать

требованиям техники безопасности, санитарным нормам и требованиям пожарной безопасности.

Требования к защите информации от несанкционированного доступа

Система должна обеспечивать защиту от несанкционированного доступа на уровне не ниже установленного требованиями, предъявляемыми к категории 1Д по классификации действующего руководящего документа «Автоматизированные системы. Защита от несанкционированного доступа к информации. Классификация автоматизированных систем».

Компоненты подсистемы защиты от несанкционированного доступа должны обеспечивать:

- идентификацию пользователя;
- проверку полномочий пользователя при работе с системой;
- разграничение доступа пользователей на уровне задач и информационных массивов.

Требования по сохранности информации при авариях

Автоматизированная система администрирования должна восстанавливать свое функционирование при корректном перезапуске аппаратных средств. Должна быть предусмотрена возможность организации автоматического и ручного резервного копирования данных системы средствами системного и базового программного обеспечения, входящего в состав программно-технического комплекса.

Требования к защите от влияния внешних воздействий

Защита от влияния внешних воздействий на АС должна обеспечиваться средствами программно-технического комплекса эксплуатирующей организации.

Требования к патентной частоте

Установка системы в целом, как и установка отдельных частей системы не должна предъявлять дополнительных требований к покупке лицензий на программное обеспечение сторонних производителей.

Требования к техническому обеспечению

Техническое обеспечение системы должно максимально и наиболее эффективным образом использовать существующие технические средства.

В состав комплекса инфраструктуры предприятия должны входить следующие технические средства:

- сервер баз данных Oracle;
- ПК администратора СЦ;
- ПК работников СЦ.

Требования к техническим характеристикам сервера баз данных:

- процессор – 2×3 ГГц;
- объем оперативной памяти – 8 Гб;
- дисковая подсистема – 900 Гб;
- устройство чтения компакт-дисков (DVD-ROM);
- сетевой адаптер – 100 Мбит.

Требования к техническим характеристикам ПК пользователя и ПК администратора:

- процессор с тактовой частотой не менее 2 ГГц;
- объем оперативной памяти – 2Гб;
- дисковая подсистема – 120 Гб;
- сетевой адаптер – 100 Мбит.

Требования к организационному обеспечению

Организационное обеспечение системы должно быть достаточным для эффективного выполнения персоналом возложенных на него обязанностей при осуществлении автоматизированных и связанных с ними неавтоматизированных функций системы. Заказчиком должны быть определены должностные лица, ответственные за:

- обработку информации;
- администрирование;
- обеспечение безопасности информации;

· управление работой персонала по обслуживанию.

К работе с системой должны допускаться сотрудники, имеющие навыки работы на персональном компьютере, ознакомленные с правилами эксплуатации.

1.5.5 Порядок контроля и приемки системы

Выпускная квалификационная работа должна включать в себя разделы, соответствующие пунктам задания на выпускную квалификационную работу. Первый этап выполнения выпускной квалификационной работы предусматривает разработку и утверждение технического задания.

1.5.6 Требования к документированию

Для системы на различных стадиях создания должны быть выпущены документы из числа предусмотренных в ГОСТ 34.201–89 в таблице 2.

Таблица 2 – Перечень разрабатываемых документов

№	Наименование документа	Часть проекта
1	Техническое задание	Формирование требований
2	Технический проект	Формирование требований
3	Руководство программиста	Ввод в действие

1.6 Выводы по главе

В результате решения поставленных задач было проведено исследование предметной области, выполнен обзор существующих программных средств аналогичного проектируемой системе назначения, сформулированы основные требования к программному продукту.

2 Проектирование и реализация информационной системы

Для реализации программного продукта, автоматизирующего учет заявок на обслуживание, необходимо спроектировать и реализовать структуру базы данных, после чего требуется разработать программное приложение для работы с этой базой данных.

Под базой данных понимается поименованная совокупность взаимосвязанных данных, находящихся под управлением системы управления базами данных (СУБД). СУБД – это комплекс программных и языковых средств, необходимых для создания баз данных, поддержания их в актуальном состоянии и организации в них информации.

Основная задача базы данных – хранить и при необходимости представлять по первому требованию пользователей все необходимые данные в одном месте, исключая их повторение и избыточность.

Технология клиент-сервер, с использованием которой будет разрабатываться приложение, подразумевает, что помимо хранения базы данных центральный компьютер (сервер базы данных) должен обеспечивать выполнение основного объема обработки данных. При использовании технологии клиент-сервер запрос на выполнение операции с данными (например, обычная выборка), выдаваемый клиентом (рабочей станцией), порождает на сервере поиск и извлечение данных. Извлеченные данные (но не файлы) транспортируются по сети от сервера к клиенту. Система, использующая технологию клиент-сервер, разделяется на две части: клиентская часть (*front-end*) обеспечивает графический интерфейс и находится на компьютере пользователя; серверная часть (*back-end*), которая находится на специально выделенных компьютерах, обеспечивает управление данными, разделение информации, администрирование и безопасность. Спецификой архитектуры клиент-сервер является использование специального языка структурированных запросов (*Structured*

Query Language, SQL), обеспечивающего пользователя простым и эффективным инструментом доступа к данным.

2.1 Логическое проектирование базы данных

При проектировании базы данных в реляционной СУБД основной целью разработки логической модели данных является создание точного представления данных, связей между ними и требуемых ограничений. Это осуществляется группировкой атрибутов в отношения так, чтобы минимизировать избыточность данных и таким образом сократить объем памяти, необходимый для физического хранения отношений, представленных в виде таблиц. Дело в том, что при работе с отношениями, содержащими избыточные данные, могут возникать проблемы, которые называются аномалиями обновления и подразделяются на аномалии вставки, удаления и модификации. Для достижения этой цели, прежде всего, необходимо определить подходящий набор отношений. Метод, который используется для решения этой задачи, называется нормализацией.

Нормализация – это формальный метод анализа отношений на основе их первичного ключа (или потенциальных ключей) и существующих функциональных зависимостей.

Он включает ряд правил, которые могут использоваться для проверки отдельных отношений таким образом, чтобы вся база данных могла быть нормализована до желаемой степени нормализации. При работе с реляционной моделью данных важно понимать, что только удовлетворение требований первой нормальной формы (1НФ) обязательно для создания отношений приемлемого качества. Все остальные формы могут использоваться по желанию проектировщиков. Однако для того чтобы избежать аномалий обновления, рекомендуется выполнять нормализацию как минимум до 3НФ.

Зачастую нормализация осуществляется в несколько последовательно выполняющихся этапов, каждый из которых соответствует некоторой нормальной форме, обладающей известными свойствами. В ходе нормализации формат отношений становится все более строгим и менее уязвимым по отношению к аномалиям обновления.

Первая нормальная форма (1НФ) – это отношение, в котором все используемые домены содержат только скалярные значения.

Ни одна из предложенных сущностей не взаимодействует сама с собой, то есть отсутствуют рекурсивные связи. Повторяющихся групп нет, т.е. все таблицы уже находятся в первой нормальной форме.

Вторая нормальная форма (2НФ) – это отношение, которое находится в первой нормальной форме и каждый атрибут которого, не входящий в состав первичного ключа, характеризуется полной функциональной зависимостью от этого первичного ключа. Вторая нормальная форма применяется к отношениям с составными ключами, т. е. к таким отношениям, первичный ключ которых состоит из двух или больше атрибутов. А так как отношение, у которого первичный ключ состоит из одного атрибута, всегда находится во второй нормальной форме. Все отношения, содержащиеся в представляемой базе, данных имеют первичные ключи на основе одного атрибута, значит можно считать, что база находится во второй нормальной форме.

Третья нормальная форма (3НФ) – это отношение, которое находится в первой и второй нормальных формах и не имеет не входящих в первичный ключ атрибутов, которые находились бы в транзитивной функциональной зависимости от этого первичного ключа.

Нормализация 2НФ-отношений с образованием 3НФ-отношений включает устранение транзитивных зависимостей. Если в отношении существует транзитивная зависимость между атрибутами, то в таком случае транзитивно-зависимые атрибуты удаляются из него и помещаются в новое отношение вместе с копией детерминанта. В представляемой базе данных не

содержатся транзитивно функционально зависимых от первичного ключа атрибутов, следовательно, база находится в 3НФ. В таблице 3 рассмотрены все таблицы базы данных и краткое описание информации, которая в них находится. Логическая модель базы данных, диаграмма компонентов и диаграмма вариантов использования представлены на рисунках 1,2 и 3.

Таблица 3 – Описание реляционных таблиц базы данных

Название таблицы	**Какая содержится информация**
D_CATEGORY	Категория поступившей заявки
D_CHANGE_LOG	История изменения заявки
D_CHRONOGRAMM	Хронология изменений в заявке
D_CLOSE_CODE	Код закрытия заявки
D_CRITICALITY	Критичность заявки
D_DEPARTMENT	Подразделение
D_EXT_CATEGORY	Категория заявки
D_GR_DISTRIBUTION	Рабочая группа
D_INFO	Информация о пользователях
D_INITIATOR	Добавление инициатора
D_OBJ	Группирование объектов
D_OBTAIN_WAY	Способ принятия заявки
D_OP_DISTRIBUTION	Таблица для связи операции с подразделением
D_OPERATION	Операции
D_PARAMS	Параметры
D_PERF_DISTRIBUTION	Таблица для связи подразделения с плательщиком
D_PERFORMER	Плательщик
D_PRIV_DISTRIBUTION	Таблица для связи подразделения с уровнем доступа
D_PRIV_GRANTS	Предоставленные полномочия
D_PRIVILEGE	Полномочия
D_REQUEST	Журнал заявок
D_ROLE	Уровень доступа к системе
D_ROLE_GRANTS	Уровни доступа к системе
D_SEARCH_PROFILE	Профили поиска заявок
D_SEARCH_TEMPLATE	Шаблоны поиска заявок
D_SERVICE	Сервис
D_SERVICE_DISTRIBUTION	Таблица для связи подразделения с сервисом
D_SERVICE_LEVEL	Уровень сервиса
D_SERVICE_OBJECT	Объект обслуживания
D_SYS_USER	Учетные данные пользователей
D_WORK_GROUP	Рабочая группа
D_SO_DISTRIBUTION	Родительский объект подразделения
D_SPEC_SERVICE	Таблица для связи сервиса со спецификациями
D_STATUS	Статус заявки
D_SV_SPECIFICATION	Спецификации
D_SYS_AGENT	Таблица для агента базы данных
INITIATOR	Инициатор

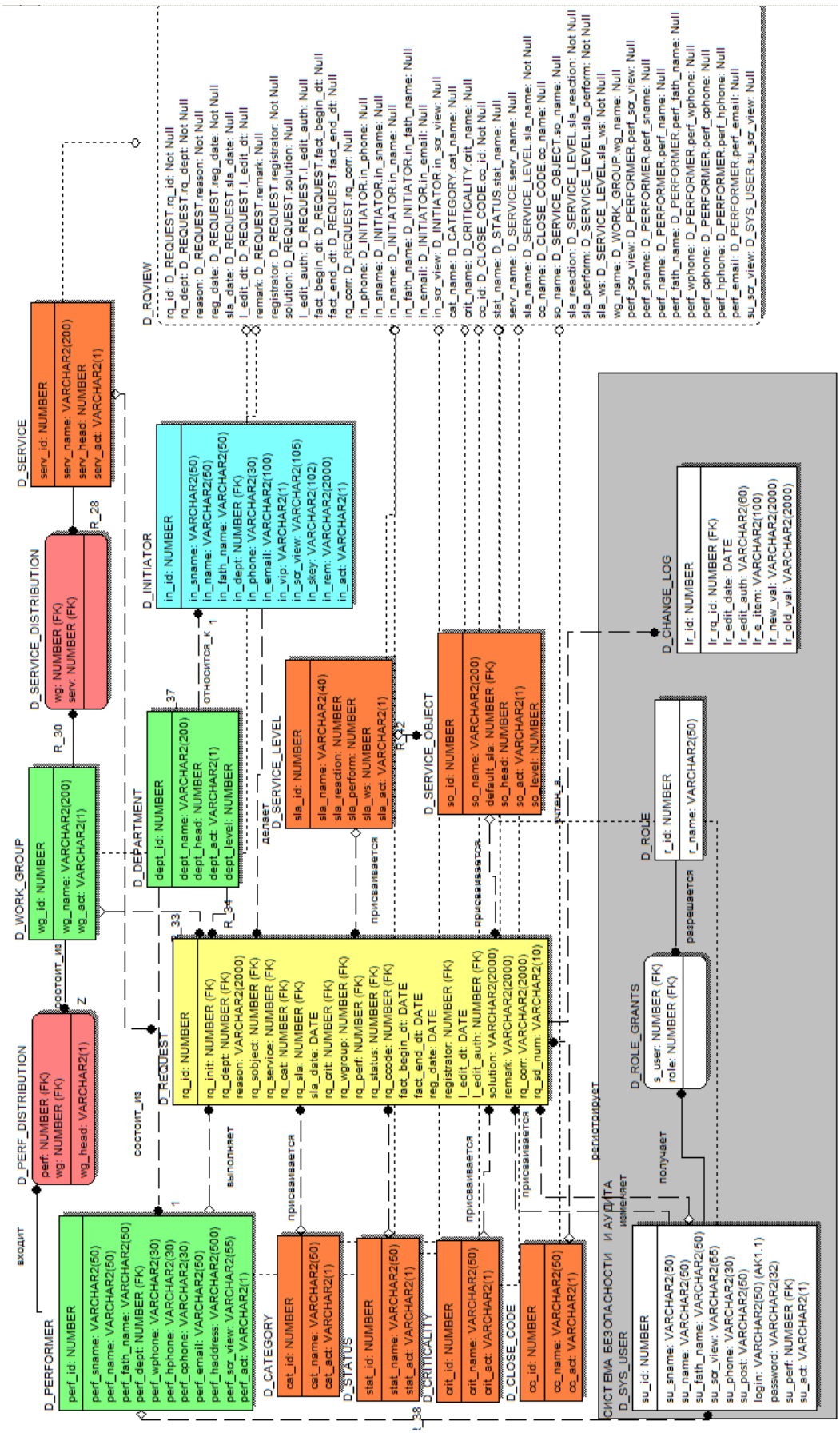

Рисунок 1 - Логическая модель базы данных

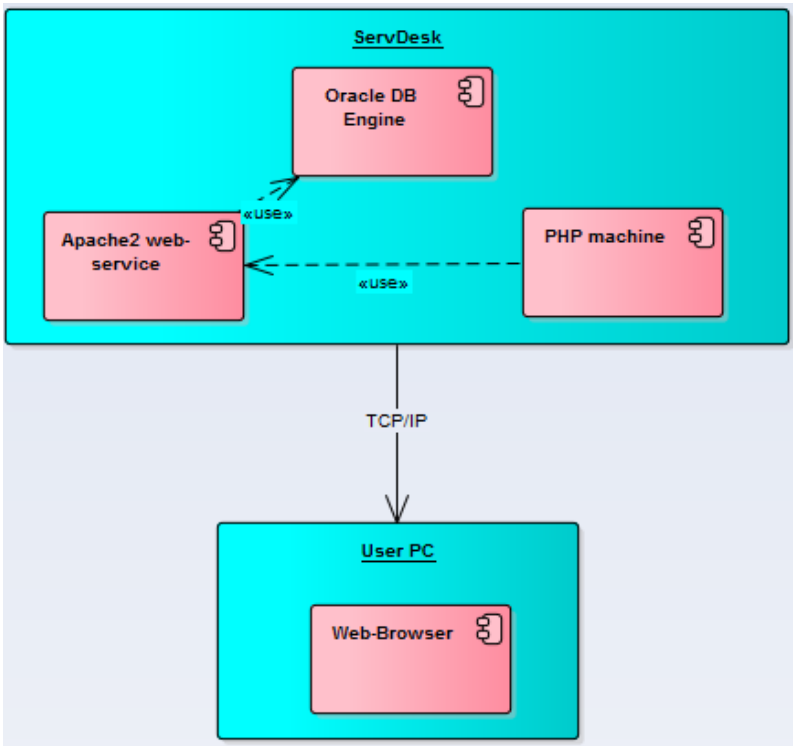

Рисунок 2 – Диаграмма компонентов

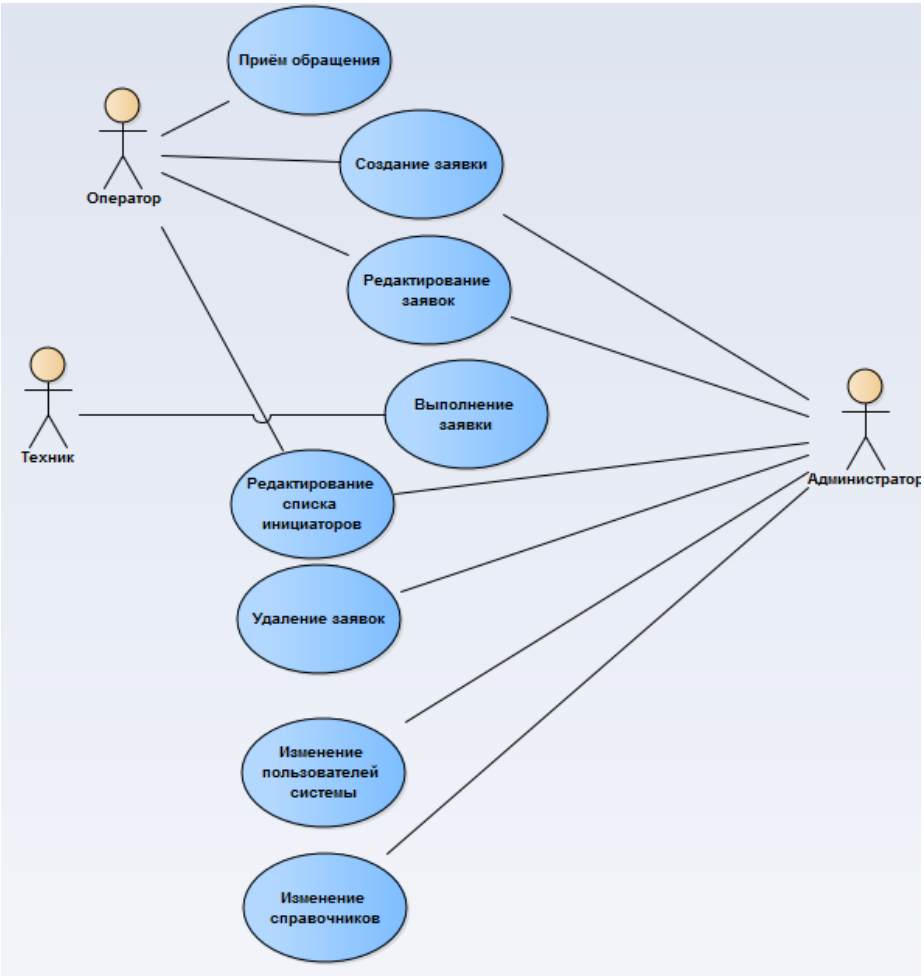

Рисунок 3 – Диаграмма вариантов использования

2.2 Физическое проектирование базы данных

Целью этого этапа является создание базовой функциональной схемы реляционной базы данных на основе глобальной логической модели данных. Также необходимо создать описание таблиц базы данных. В таблице 3 представлена физическая модель базы данных.

Таблица 3 – Описание таблиц базы данных

Field		Type	Extra
D_CATEGORY			
P	CAT_ID	NUMBER	
	CAT_NAME	VARCHAR2(50 BYTE)	Allow Null
	CAT_ACT	VARCHAR2(1 BYTE)	Allow Null
D_CHANGE_LOG			
P	LR_ID	NUMBER	
	LR_RQ_ID	NUMBER	
	LR_EDIT_DATE	DATE	Allow Null
	LR_EDIT_AUTH	VARCHAR2(60 BYTE)	Allow Null
	LR_E_ITEM	VARCHAR2(100 BYTE)	Allow Null
	LR_NEW_VAL	VARCHAR2(2000 BYTE)	Allow Null
	LR_OLD_VAL	VARCHAR2(2000 BYTE)	Allow Null
	LR_NEW_SPVAL	NUMBER	Allow Null
	LR_OLD_SPVAL	NUMBER	Allow Null
D_CHRONOGRAMM			
P	CHRONO_ID	NUMBER	
	CHRONO_RQID	NUMBER	
	CHRONO_DATE	DATE	
	CHRONO_CSTATUS	NUMBER	
	CHRONO_PSTATUS	NUMBER	Allow Null
	CHRONO_AUTHOR	NUMBER	
D_CLOSE_CODE			
P	CC_ID	NUMBER	
	CC_NAME	VARCHAR2(50 BYTE)	Allow Null
	CC_ACT	VARCHAR2(1 BYTE)	Allow Null
D_CRITICALITY			
P	CC_ID	NUMBER	
	CC_NAME	VARCHAR2(50 BYTE)	Allow Null
	CC_ACT	VARCHAR2(1 BYTE)	Allow Null
D_DEPARTMENT			
P	DEPT_ID	NUMBER	
	DEPT_NAME	VARCHAR2(200 BYTE)	Allow Null
	DEPT_HEAD	NUMBER	Allow Null
	DEPT_ACT	VARCHAR2(1 BYTE)	Allow Null
	DEPT_LEVEL	NUMBER	Allow Null
D_EXT_CATEGORY			
P	ECAT_ID	NUMBER	
	ECAT_NAME	VARCHAR2(50 BYTE)	

Продолжение таблицы 3

	Field	Type	Extra
	ECAT_ACT	VARCHAR2(1 BYTE)	
	D_GR_DISTRIBUTION		
P	GRD_WGROUP	NUMBER	
P	GRD_DEPT	NUMBER	
	D_INFO		
P	INFO_ID	NUMBER	
	SU_ID	NUMBER	Allow Null
	INFO_DATA	VARCHAR2(500 BYTE)	Allow Null
	INFO_NEW	VARCHAR2(1 BYTE)	Allow Null
	INFO_TYPE	VARCHAR2(1 BYTE)	Allow Null
	D_INITIATOR		
P	IN_ID	NUMBER	
	IN_SNAME	VARCHAR2(50 BYTE)	Allow Null
	IN_NAME	VARCHAR2(50 BYTE)	Allow Null
	IN_FATH_NAME	VARCHAR2(50 BYTE)	Allow Null
	IN_DEPT	NUMBER	
	IN_PHONE	VARCHAR2(30 BYTE)	Allow Null
	IN_EMAIL	VARCHAR2(100 BYTE)	Allow Null
	IN_VIP	VARCHAR2(1 BYTE)	Allow Null
	IN_SCR_VIEW	VARCHAR2(105 BYTE)	Allow Null
	IN_SKEY	VARCHAR2(102 BYTE)	Allow Null
	IN_REM	VARCHAR2(2000 BYTE)	Allow Null
	IN_ACT	VARCHAR2(1 BYTE)	Allow Null
	IN_JOBSITE	VARCHAR2(200 BYTE)	Allow Null
	D_OBJ		
P	OBJ_ID	NUMBER	
	OBJ_PATCH	VARCHAR2(50 BYTE)	Allow Null
	OBJ_DESCRIPTION	VARCHAR2(250 BYTE)	Allow Null
	OBJ_POSITION	NUMBER	Allow Null
	D_OBTAIN_WAY		
P	OW_ID	NUMBER	
	OW_NAME	VARCHAR2(50 BYTE)	Allow Null
	OW_ACT	VARCHAR2(1 BYTE)	
	D_OP_DISTRIBUTION		
	OPD_OPERATION	NUMBER	
	OPD_SOBJECT	NUMBER	
	D_OPERATION		
P	OP_ID	NUMBER	
	OP_NAME	VARCHAR2(200 BYTE)	Allow Null
	OP_LOCATION	NUMBER	
	OP_ACT	VARCHAR2(1 BYTE)	Allow Null
	D_PARAMS		
P	PARAM	VARCHAR2(100 BYTE)	
	VALUE	VARCHAR2(500 BYTE)	Allow Null
	REM	VARCHAR2(500 BYTE)	Allow Null
	D_PERF_DISTRIBUTION		
P	PERF	NUMBER	
P	WG	NUMBER	
	WG_HEAD	VARCHAR2(1 BYTE)	Allow Null
	D_PERFORMER		

Продолжение таблицы 3

Field		Type	Extra
P	PERF_ID	NUMBER	
	PERF_SNAME	VARCHAR2(50 BYTE)	Allow Null
	PERF_NAME	VARCHAR2(50 BYTE)	Allow Null
	PERF_FATH_NAME	VARCHAR2(50 BYTE)	Allow Null
	PERF_DEPT	NUMBER	
	PERF_WPHONE	VARCHAR2(30 BYTE)	Allow Null
	PERF_HPHONE	VARCHAR2(30 BYTE)	Allow Null
	PERF_CPHONE	VARCHAR2(30 BYTE)	Allow Null
	PERF_EMAIL	VARCHAR2(50 BYTE)	Allow Null
	PERF_HADDRESS	VARCHAR2(500 BYTE)	Allow Null
	PERF_SCR_VIEW	VARCHAR2(55 BYTE)	Allow Null
	PERF_ACT	VARCHAR2(1 BYTE)	Allow Null
D_PRIV_DISTRIBUTION			
P	PD_PRIVILEGE	NUMBER	
P	PD_ROLE	NUMBER	
D_PRIV_GRANTS			
P	PG_SUSERID	NUMBER	
P	PG_SUSERID	NUMBER	
D_PRIVILEGE			
P	PR_ID	NUMBER	
	PR_NAME	VARCHAR2(250 BYTE)	
D_REQUEST			
P	D_REQUEST	NUMBER	
	RQ_SD_NUM	VARCHAR2(20 BYTE)	Allow Null
	RQ_PUB_NUM	VARCHAR2(20 BYTE)	
	RQ_INIT	NUMBER	
	RQ_DEPT	NUMBER	
	REASON	VARCHAR2(2000 BYTE)	
	RQ_SOBJECT	NUMBER	Allow Null
	RQ_SERVICE	NUMBER	Allow Null
	RQ_CAT	NUMBER	
	RQ_SLA	NUMBER	Allow Null
	SLA_DATE	DATE	Allow Null
	RQ_CRIT	NUMBER	Allow Null
	RQ_WGROUP	NUMBER	Allow Null
	RQ_PERF	NUMBER	Allow Null
	RQ_STATUS	NUMBER	
	RQ_CCODE	NUMBER	Allow Null
	FACT_BEGIN_DT	DATE	Allow Null
	FACT_END_DT	DATE	Allow Null
	REG_DATE	DATE	
	REGISTRATOR	NUMBER	
	L_EDIT_DT	DATE	Allow Null
	L_EDIT_AUTH	NUMBER	Allow Null
	SOLUTION	VARCHAR2(2000 BYTE)	Allow Null
	REMARK	VARCHAR2(2000 BYTE)	Allow Null
	RQ_CORR	VARCHAR2(2000 BYTE)	Allow Null
	RQ_ATTACH	VARCHAR2(1 BYTE)	Allow Null
	RQ_CONFIRM	VARCHAR2(1 BYTE)	Allow Null

Продолжение таблицы 3

	Field	Type	Extra
	RQ_NEGOTIATOR	NUMBER	Allow Null
	RQ_OPERATION	NUMBER	Allow Null
	RQ_DURATION	NUMBER	Allow Null
	RQ_INNOTE	VARCHAR2(1 BYTE)	Allow Null
	RQ_OWAY	NUMBER	Allow Null
	RQ_CONFIRM_DT	DATE	Allow Null
	RQ_ECAT	NUMBER	Allow Null
	D_ROLE		
P	R_ID	NUMBER	
	R_NAME	VARCHAR2(50 BYTE)	Allow Null
	D_ROLE_GRANTS		
P	ROLE	NUMBER	
P	S_USER	NUMBER	
	D_SEARCH_PROFILE		
P	SU_ID	NUMBER	
P	ST_ID	NUMBER	
	DEFAULT_ST	VARCHAR2(1 BYTE)	Allow Null
	REFRESH_INTERVAL	NUMBER	Allow Null
	D_SEARCH_TEMPLATE		
P	ST_ID	NUMBER	
	ST_NAME	VARCHAR2(100 BYTE)	Allow Null
	ST_DATA	VARCHAR2(2000 BYTE)	Allow Null
	ST_TYPE	VARCHAR2(1 BYTE)	Allow Null
	ST_OWNER	NUMBER	Allow Null
	D_SERVICE		
P	SERV_ID	NUMBER	
	SERV_NAME	VARCHAR2(200 BYTE)	Allow Null
	SERV_HEAD	NUMBER	Allow Null
	SERV_ACT	VARCHAR2(1 BYTE)	Allow Null
	SERV_LEVEL	NUMBER	Allow Null
	SERV_PATH	VARCHAR2(180 BYTE)	Allow Null
	SERV_GP	VARCHAR2(1 BYTE)	Allow Null
	D_SERVICE_DISTRIBUTION		
P	WG	NUMBER	
P	SERV	NUMBER	
	D_SERVICE_LEVEL		
P	SLA_ID	NUMBER	
	SLA_NAME	VARCHAR2(40 BYTE)	
	SLA_REACTION	NUMBER	
	SLA_PERFORM	NUMBER	
	SLA_WS	NUMBER	
	SLA_ACT	VARCHAR2(1 BYTE)	Allow Null
	D_SERVICE_OBJECT		
P	SO_ID	NUMBER	
	SO_NAME	VARCHAR2(200 BYTE)	Allow Null
	DEFAULT_SLA	NUMBER	Allow Null
	SO_HEAD	NUMBER	Allow Null
	SO_ACT	VARCHAR2(1 BYTE)	Allow Null
	SO_LEVEL	NUMBER	Allow Null
	SERV_ID	NUMBER	Allow Null

Продолжение таблицы 3

	Field	Type	Extra
	SO_ISGROUP	VARCHAR2(1 BYTE)	Allow Null
		D_SO_DISTRIBUTION	
	SOD_SOBJECT	NUMBER	
	SOD_WGROUP	NUMBER	
		D_SPEC_SERVICE	
P	SPCSRV_ID	NUMBER	
	SPCSRV_NAME	VARCHAR2(2000 BYTE)	Allow Null
		D_STATUS	
P	STAT_ID	NUMBER	
	STAT_NAME	VARCHAR2(50 BYTE)	Allow Null
	STAT_ACT	VARCHAR2(1 BYTE)	Allow Null
	STAT_ORDER	NUMBER	Allow Null
		D_SV_SPECIFICATION	
P	SOBJ_ID	NUMBER	
P	SPCS_ID	NUMBER	
P	SL_ID	NUMBER	
		D_SYS_AGENT	
P	AGENT_ID	NUMBER	
	AGENT_PERFID	NUMBER	Allow Null
	AGENT_INITID	NUMBER	Allow Null
	AGENT_SUID	NUMBER	Allow Null
	AGENT_DEPT	NUMBER	
	AGENT_IS_PERFORMER	VARCHAR2(1 BYTE)	
	AGENT_IS_INITIATOR	VARCHAR2(1 BYTE)	
	AGENT_IS_VIP	VARCHAR2(1 BYTE)	
	AGENT_INI_TYPE	VARCHAR2(1 BYTE)	
	AGENT_PERF_ACT	VARCHAR2(1 BYTE)	
	AGENT_INI_ACT	VARCHAR2(1 BYTE)	
	AGENT_ACT	VARCHAR2(1 BYTE)	
	AGENT_SNAME	VARCHAR2(50 BYTE)	Allow Null
	AGENT_NAME	VARCHAR2(50 BYTE)	Allow Null
	AGENT_FNAME	VARCHAR2(50 BYTE)	Allow Null
	AGENT_HPHONE	VARCHAR2(30 BYTE)	Allow Null
	AGENT_WPHONE	VARCHAR2(30 BYTE)	Allow Null
	AGENT_CPHONE	VARCHAR2(30 BYTE)	Allow Null
	AGENT_EMAIL	VARCHAR2(100 BYTE)	Allow Null
	AGENT_SCR_VIEW	VARCHAR2(55 BYTE)	Allow Null
	AGENT_POST	VARCHAR2(150 BYTE)	Allow Null
	AGENT_SKEY	VARCHAR2(50 BYTE)	
	AGENT_JOBSITE	VARCHAR2(200 BYTE	Allow Null
	AGENT_HADDRESS	VARCHAR2(2000 BYTE)	Allow Null
	AGENT_REM	VARCHAR2(2000 BYTE)	Allow Null
	AGENT_LOGIN	VARCHAR2(60 BYTE)	Allow Null
	AGENT_PASS_HASH	VARCHAR2(32 BYTE)	Allow Null
		D_SYS_USER	
P	SU_ID	NUMBER	
	SU_SNAME	VARCHAR2(50 BYTE)	Allow Null
	SU_NAME	VARCHAR2(50 BYTE)	Allow Null
	SU_FATH_NAME	VARCHAR2(50 BYTE)	

Окончание таблицы 3

Field		Type	Extra
	SU_SCR_VIEW	VARCHAR2(55 BYTE)	
	SU_PHONE	VARCHAR2(30 BYTE)	
	SU_POST	VARCHAR2(50 BYTE)	Allow Nul
	LOGIN	VARCHAR2(50 BYTE)	
	PASSWORD	VARCHAR2(32 BYTE)	Allow Nul
	SU_PERF	NUMBER	Allow Null
	SU_ACT	VARCHAR2(1 BYTE)	Allow Null
D_WORK_GROUP			
P	WG_ID	NUMBER	
	WG_NAME	VARCHAR2(200 BYTE)	Allow Null
	WG_ACT	VARCHAR2(1 BYTE)	Allow Null
	WG_PARENT	NUMBER	Allow Null
INITIATOR			
	IN_SNAME	VARCHAR2(50 BYTE)	Allow Null
	IN_NAME	VARCHAR2(50 BYTE)	Allow Null
	IN_FATH_NAME	VARCHAR2(50 BYTE)	Allow Null
	DEPT_NAME	VARCHAR2(200 BYTE)	Allow Null
	DEPT_FULL_NAME	VARCHAR2(4000 BYTE)	Allow Null
	COMPANY_ID	NUMBER	Allow Null
	IN_PHONE	VARCHAR2(30 BYTE)	Allow Null
	IN_EMAIL	VARCHAR2(100 BYTE)	Allow Null
	VIP	VARCHAR2(1 BYTE)	Allow Null
	IN_SCR_VIEW	VARCHAR2(105 BYTE)	Allow Null
	IN_REM	VARCHAR2(2000 BYTE)	Allow Null
	DOL	VARCHAR2(500 BYTE)	Allow Null
	TEL	VARCHAR2(100 BYTE)	Allow Null

2.3 Серверная часть

Oracle Database Express Edition (*Oracle Database XE*) - это система управления базами данных (СУБД) начального уровня, основанная на программном коде СУБД *Oracle Database*. *Oracle Database XE* может быть установлена на любой компьютер с любым количеством процессоров (одна база данных на машину), но с ограничением в 11Гб пользовательских данных, использует не более 1Гб оперативной памяти и только один из имеющихся процессоров. Ограничения, накладываемые на версию *Oracle Database XE,* могут различаться , так например, в версии *Oracle Database XE* 10*g*, максимальный объем пользовательских данных был равен 4Гб, а в *Oracle Database XE* 11*g* -11Гб. Кроме всего прочего, в *XE*-версии, по

сравнению с платными продуктами, отсутствуют расширенные средства обеспечения безопасности, технология шифрования *SecureFiles Encryption*, функции для повышения производительности, оптимизации и надежного хранения больших объемов данных. Тем не менее, *Oracle Database XE* представляет собой отличную СУБД, вполне пригодную для:

- разработчиков, работающих с *PHP*, Java, *.NET, XML* и приложениями с открытым кодом;
- администраторов баз данных, которым нужна небольшая бесплатная СУБД;
- независимых вендоров, которые хотят бесплатную СУБД стартового уровня для продажи без лишних затрат;
- учебных заведений и студентов, которым необходима бесплатная СУБД.

Для работы с *Oracle Database XE* используется обычный браузер, обеспечивающий простой и удобный *web*-интерфейс, позволяющий:

- администрировать базу данных;
- создавать таблицы, представления и другие объекты базы данных;
- импортировать, экспортировать и просматривать табличные данные;
- выполнять запросы и запускать *SQL*-скрипты;
- генерировать отчеты.

В случае необходимости, имеется возможность простого и быстрого перевода СУБД на коммерческую платформу без потери пользовательских данных.

2.3.1 Установка *Oracle Database Express Edition*

Требования к оборудованию и операционным системам для установки *Oracle Express Edition* имеются в документации к приложению. Необходимо отметить, что список операционных систем семейства *Linux*, рекомендуемый разработчиком для установки *Oracle Database* 11*g Express Edition* довольно скромный:

- *Oracle Enterprise Linux* 4 *Update* 7;
- *Oracle Enterprise Linux* 5 *Update* 2;
- *Red Hat Enterprise Linux* 4 *Update* 7;
- *Red Hat Enterprise Linux* 5 Update 2;
- *SUSE Linux Enterprise Server* 10 *SP*2;
- *SUSE Linux Enterprise Server* 11.

Тем не менее, установка например *Oracle Database XE* 11*g* возможна и в среде других ОС семейства *Linux* при соответствии минимальным требованиям к системе, в частности - с версией ядра не ниже 2.4.21 и при наличии следующих установленных пакетов:

- *glibc* версии 2.3.4-2.41;
- *make* версии не ниже 3.80;
- *binutils* версии не ниже 2.16.91.0.5;
- *gcc* версии 4.1.2 или старше;
- *libaio* версии 0.3.104 или старше.

Перед установкой *Oracle Database XE* необходимо удалить предыдущую версию СУБД, если она была установлена на данной системе.

Для установки *Oracle Database* XE нужно:

1. Войти в систему с правами *root*.

2. Перейти на страницу *Oracle Database XE*.

3. Щелкнуть на ссылку *Free Download* и следовать инструкциям по загрузке *Linux* — версии *Oracle Database XE*.

4. Запустить установку *Oracle Database XE* например командой: *# rpm -ivh /downloads/oracle-xe-11.2.0-1.0.x86_64.rpm*.

5. По запросу установщика выполнить команду для настройки параметров *# /etc/init.d/oracle-xe configure*.

6. Ввести требуемые конфигурационные параметры

Для запуска службы в ручном режиме используется команда, выполняемая под учетной записью root:

/etc/init.d/oracle-xe start

Для остановки:

/etc/init.d/oracle-xe stop

В инсталляционном архиве *Oracle Database XE* 11*g* помимо пакета .rpm содержится еще 2 папки :

- *responce* - средство для выполнения «тихой установки». Содержит файл *xe.rsp*, содержимое которого используется для определения параметров установки *Oracle Database XE*;
- *upgrade* - средство для миграции данных с версии 10.2*XE* на версию 11.2*XE*. Представляет собой скрипт для выполнения в *SQLPlus* под учетной записью *SYS*.

Для выполнения ”тихой” установки нужно:

1. В файле *xe.rsp* задать параметры установки.

2. Создать скрипт для запуска установки с использованием файла *xe.rsp*.

3. Выполнить данный скрипт под учетной записью *root*. Информация о ходе установки записывается в файл журнала *XEsilentinstall.log*.

2.3.2 Установка переменных окружения для *Oracle Database Express Edition*

После завершения установки *Oracle Database XE* необходимо установить требуемые значения переменных окружения для всех пользователей, которые будут использовать СУБД. Для этого не требуются права суперпользователя, и в каталоге исполняемых файлов */u01/app/oracle/product/11.2.0/xe/bin* имеются готовые скрипты:

- *oracle_env.csh* - для оболочки *C* и *tcsh*;
- *oracle_env.sh* - для оболочек *sh* и *bash*.

Можно отредактировать файлы профилей таким образом, чтобы значения переменных устанавливались при входе пользователей в систему.

Для работы с *Oracle Database*, пользователь должен быть членом группы dba, которая создается при установке СУБД.

2.3.3 Удаленный доступ к Oracle Database Express Edition

Для работы с СУБД используется меню «Приложения» (*Applications*) – «*Oracle Database 11g Express Edition*». Способ запуска приложений изображен на рисунке 4.

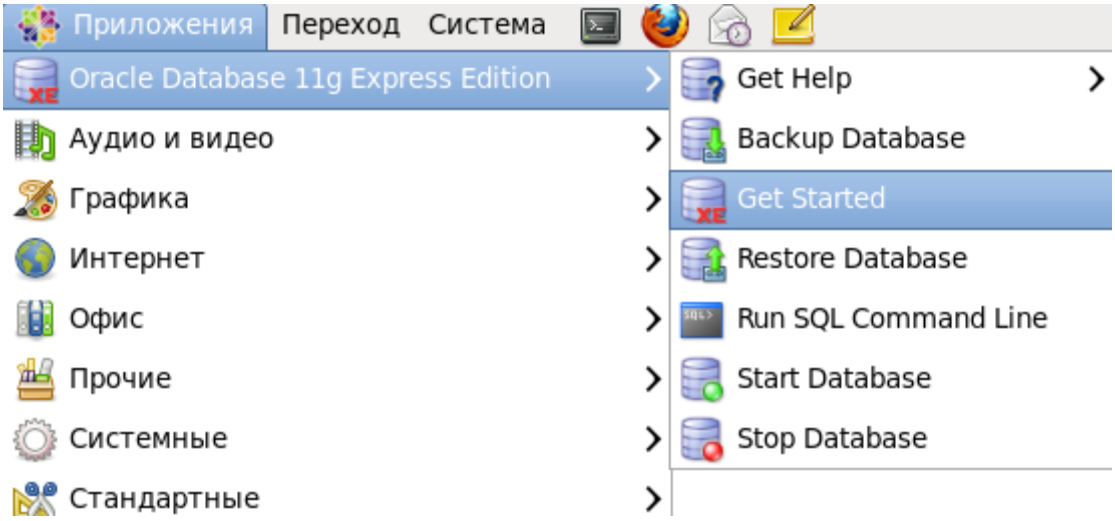

Рисунок 4 - Запуск оснастки для управления БД

- *Get Help* - страница документации;
- *Backup Database* - создание резервных копий базы данных;
- *Get Started* - стартовая страница *Oracle Database XE*. Это основное приложение для администрирования СУБД;
- *Restore Darabase* - восстановление базы данных из резервной копии;
- *Run SQL Command Line* - работа с СУБД в командной строке приложения *sqlplus*;
- *Start Database* - запустить базу данных;
- *Stop Database* - остановить базу данных.

Для администрирования *Oracle Database XE* используется «Get Started». Окно приложения показано на рисунке 5.

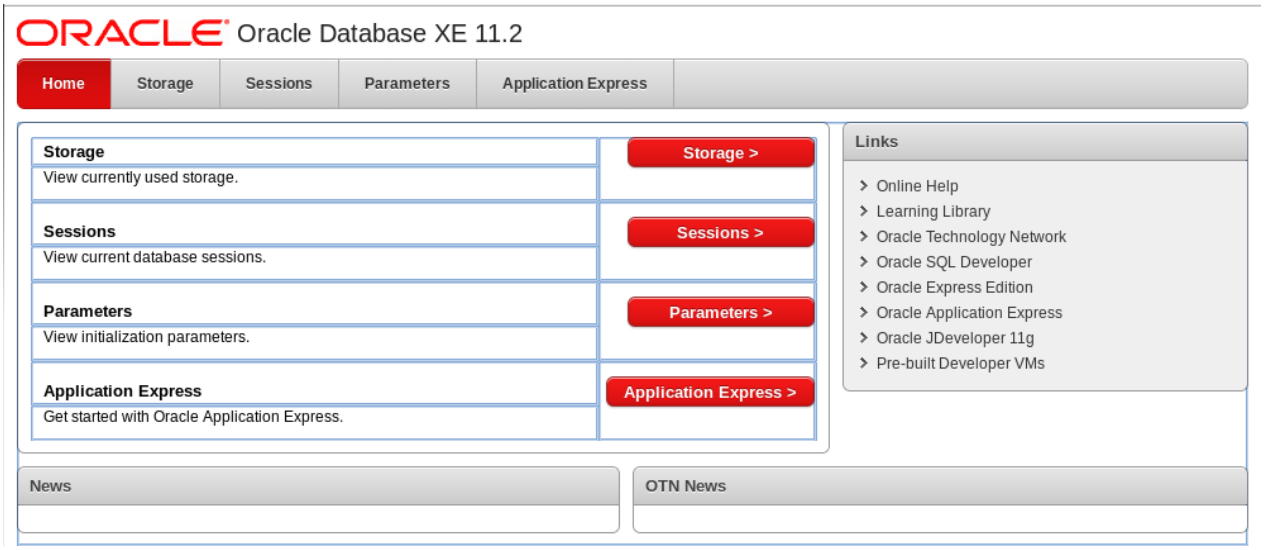

Рисунок 5- Окно управления СУБД

- *Storage* - просмотр информации об использовании памяти;
- *Sessions* - сведения о сеансах пользователей;
- *Parameters* - параметры работы СУБД;
- *Application Express* - основное приложение для работы с *Oracle Database XE*. Используется для добавления или удаления пользователей, управления службами, мониторинга активности, работы с данными и приложениями.

После установки, доступ к веб-странице для работы с СУБД возможен только на локальном компьютере. Чтобы включить удаленный доступ нужно запустить от имени суперпользователя, приложение для работы с *Oracle Database* в командной строке и подключиться к базе данных под учетной записью *SYSTEM* и выполнить следующую команду: «*EXEC DBMS_XDB.SETLISTENERLOCALACCESS(FALSE)*».

После этого возможно удаленное подключение к базам данных с использованием браузера.

Открыв окно удаленного управления, необходимо ввести имя базы данных, имя пользователя и пароль для подключения к серверу, после чего откроется основное окно приложения для работы с СУБД *Oracle Database XE* (рисунок 6).

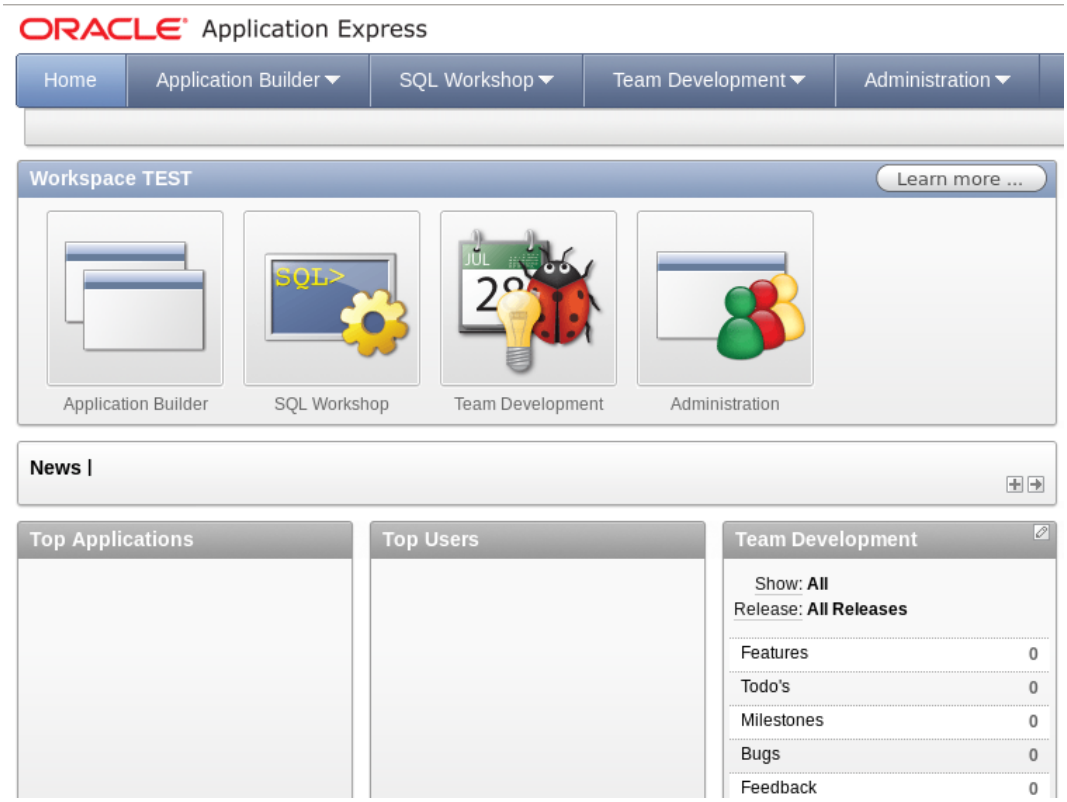

Рисунок 6 - Окно удаленного управления БД

Необходимо учитывать, что в бесплатной версии *Oracle Database XE*, данные между клиентом и сервером пересылаются в обычном текстовом виде без шифрования, и, если требуется надежная защита информации, то нужно либо выполнить переход на платную версию, либо применять стороннее программное обеспечение.

2.4 Руководство пользователя

Аутентификация выполняется посредством ввода, зарегистрированным пользователем, своего учетного имени и пароля в поля «*Логин*» и «*Пароль*» соответственно, и последующим нажатием кнопки «*Войти*».

Элементы управления необходимые для обработки данных аутентификации представлены на рисунке 7.

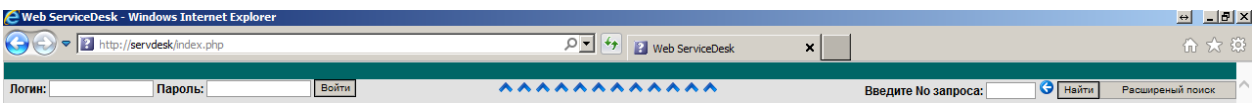

Рисунок 7 - Блок ввода данных авторизации

Данные о зарегистрированных в БДЗ системы запросах, полученные в результате работы механизмов поиска, отображаются в рабочей области в виде таблицы, содержащей:

- заголовок со служебными данными и указанием расположения информационных полей;
- тело таблицы с перечнем найденных данных (либо сообщение о пустом результате поиска запросов).

Рабочая область web-приложения содержащая таблицу общего просмотра с комментариями показана на рисунке 8.

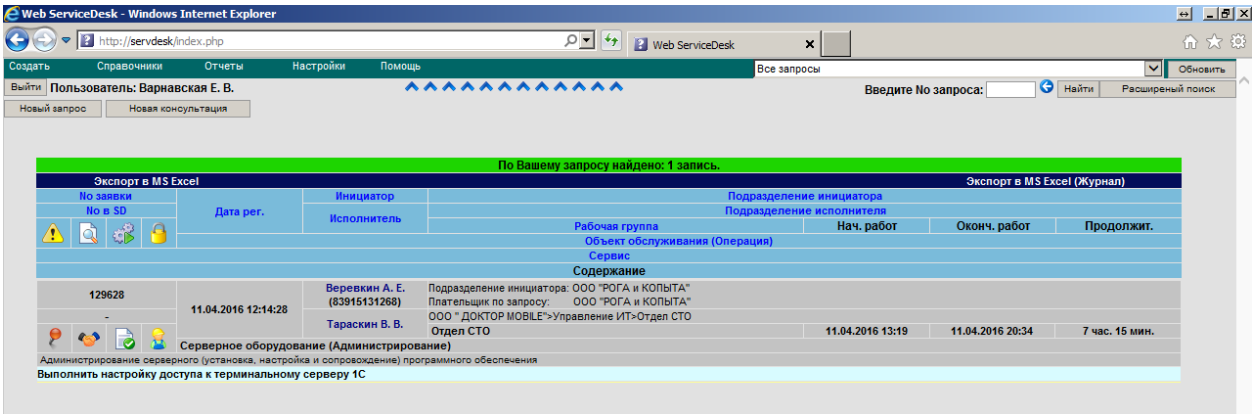

Рисунок 8 - Рабочая область web-приложения

Заголовок таблицы просмотра данных о запросах представлен на рисунке 9.

Рисунок 9 - Заголовок таблицы

в шапке заголовка расположены следующие кнопки, оформленные в виде гиперссылок (рисунок 10):

- кнопка «Экспорт в *MS Excel*» - переход по данной гиперссылке обеспечивает вывод отображаемой информации файл формата *Microsoft Excel* (**.xls*);

- кнопка «Экспорт в *MS Excel* (Журнал)» - переход по данной гиперссылке обеспечивает выгрузку отображаемой информации в файл формата *Microsoft Excel* (*.*xls*).

Указатели на основные параметры классификации запросов, отображаемые в заголовке таблицы, представлены графически - в виде служебных значков

Описание значения представленных графически заголовков полей приведены в таблице 4.

Таблица 4 - Графическое обозначение заголовков полей

ЗНАЧОК	ЗНАЧЕНИЕ
1	2
	Поле «Критичность запроса»
	Поле «Категория запроса»
	Поле «Статус запроса»
	Поле «Код закрытия»

Расшифровка графического представления значений хранимых в полях «*Критичность*», «*Категория*», «*Статус*», «*Код закрытия*» представлена в таблице 5.

Таблица 5 - Графическое представление значений параметров

ЗНАЧОК	ЗНАЧЕНИЕ
ЗНАЧЕНИЯ ПОЛЯ "КРИТИЧНОСТЬ ЗАПРОСА"	
	Очень низкая
	Обычная
	Высокая
	Очень высокая
ЗНАЧЕНИЯ ПОЛЯ "КАТЕГОРИЯ ЗАПРОСА"	
	Сбой
	Услуга

Окончание таблицы 5

ЗНАЧОК	ЗНАЧЕНИЕ
	Запрос на изменение
ЗНАЧЕНИЕ ПОЛЯ "СТАТУС ЗАПРОСА"	
	Зарегистрирован и назначен
	Принят к исполнению
	Выполнен
	Закрыт
	Отменен
ЗНАЧЕНИЕ ПОЛЯ "КОД ЗАКРЫТИЯ ЗАПРОСА"	
	Выполнен дистанционно
	Выполнен на рабочем месте пользователя
	Отменен
	Передан третьей стороне
	Решен во время обращения
	Снят заявителем
	Значение не определено

Для того чтобы перейти к режиму редактирования данных запроса необходимо навести мышь на бланк предварительного просмотра того запроса, данные о котором необходимо изменить и дважды щелкнуть на область где отображается содержание запроса

После выполнения указанных действий будет открыта форма карточки запроса в рабочем режиме, позволяющем редактирование данных выбранного запроса.

Для уточнения причины обращения либо добавление дополнительной информации о запросе следует изменить либо внести содержимое текстовых полей «*Причина обращения*» и «*Дополнительная информация*» (рисунок 8).

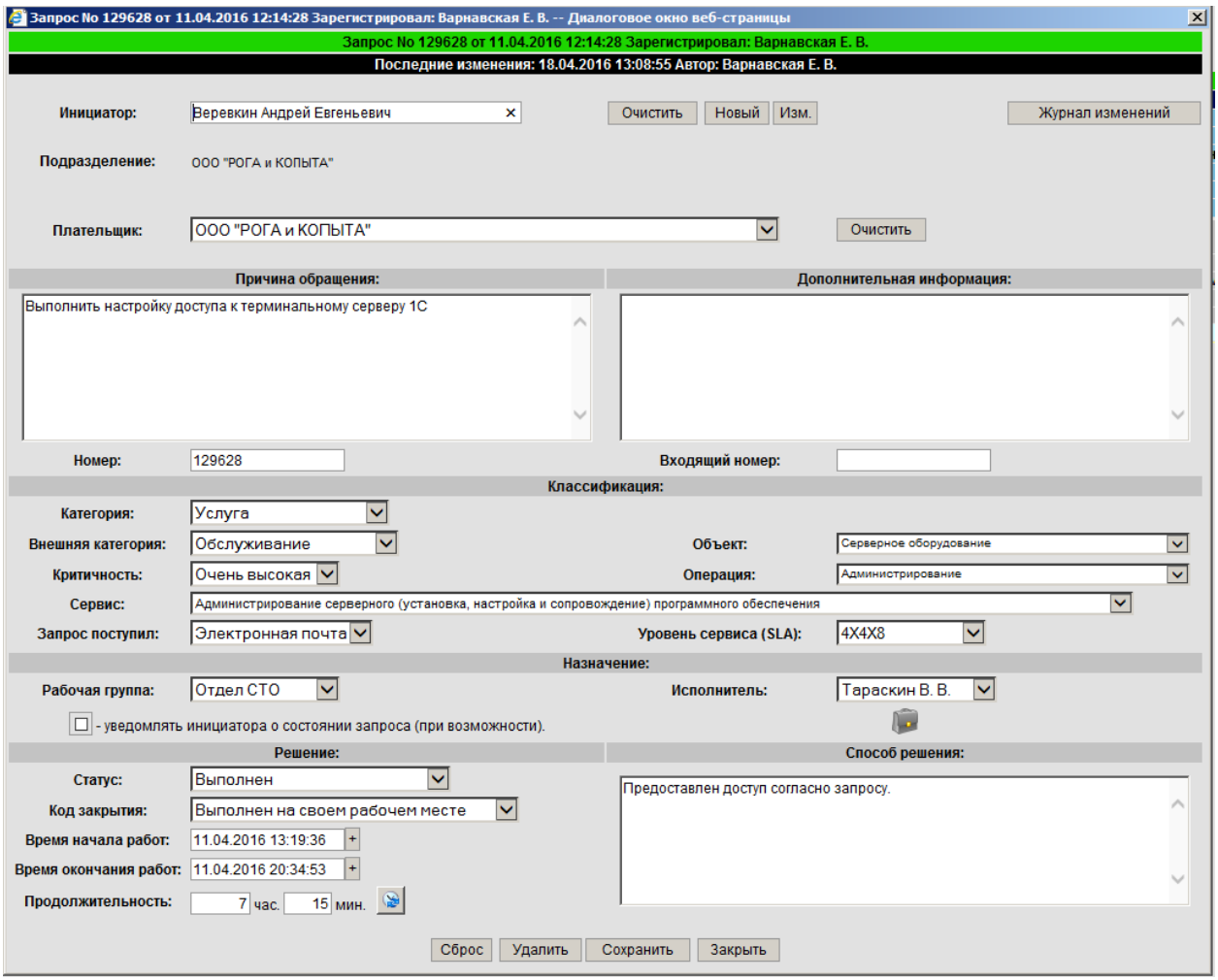

Рисунок 10 - Форма карточки запроса в режиме изменения,
поля «Причина обращения» и «Дополнительная информация»

При уточнении классификации запроса следует изменить значения полей входящих в блок «*Классификация*», изменение значений входящих в этот блок сводится к выбору значений из соответствующих выпадающих списков.

При переназначении запроса, следует изменять значение полей входящих в блок «*Назначение*» путем выбора нужного значения из соответствующих выпадающих списков.

При изменении значения поля «*Рабочая группа*», если действующий на текущий момент исполнитель, указанный в поле «*Исполнитель*», состоит в рабочей группе, на которую указывает новое значение поля «*Рабочая группа*», то значение поля «*Исполнитель*» изменяться не будет. Если действующий на текущий момент Исполнитель не состоит в рабочей группе,

на которую указывает новое значение поля «*Рабочая группа*», то автоматически в поле «*Исполнитель*» будет указан руководитель выбранной рабочей группы.

Если при регистрации запроса, к нему были прикреплены какие либо файлы, то в блоке «*Назначение*» будет отображаться иконка «». Для того чтобы просмотреть перечень прикрепленных к запросу файлов, необходимо навести мышь на иконку «», и нажать левую кнопку. По нажатию будет открыта форма диалога работы с прикрепляемыми файлами , который будет отображать список прикрепленных файлов, где перечисленные наименования файлов являются гиперссылками. При необходимости, файлы из списка можно загрузить на локальный компьютер, для этого требуется выбрать соответствующую гиперссылку из списка и выполнить переход по ней, для начала загрузки файла (рисунок 11).

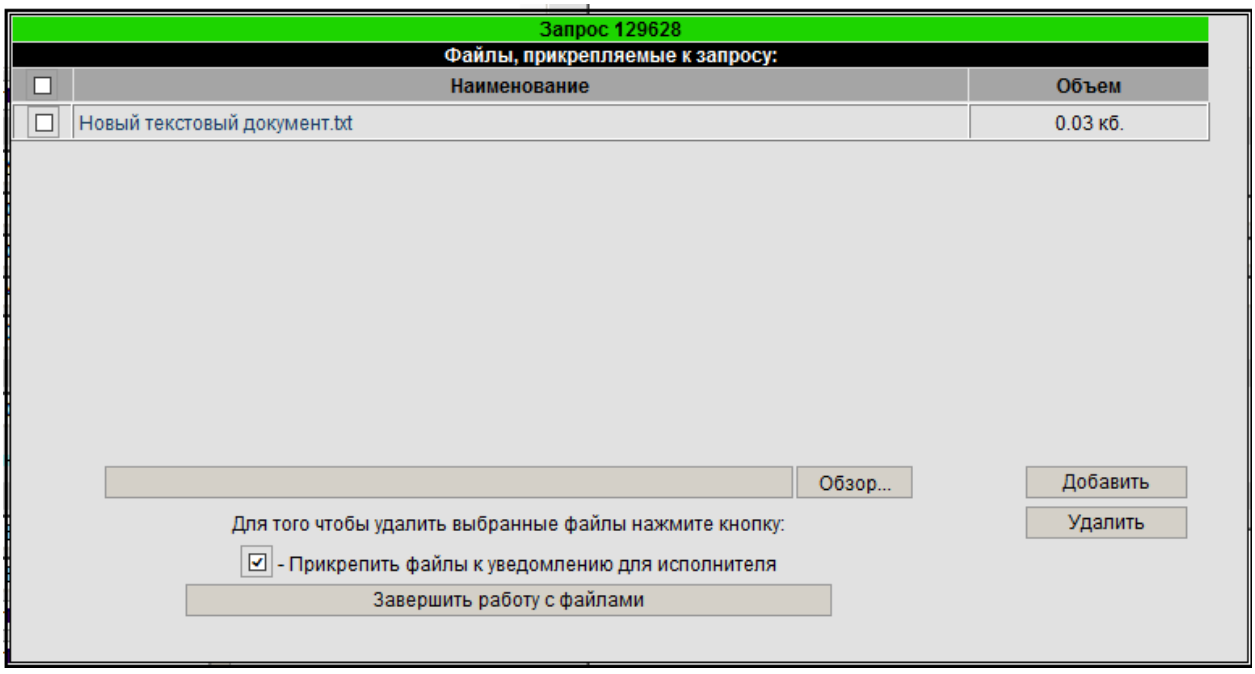

Рисунок 11 - Диалог работы с прикрепленными файлами

При изменении состояния запроса в ходе его выполнения, следует изменять значения полей входящих в блок «*Решение*» путем выбора значений из выпадающего списка соответствующих полей, ввода дат

вручную, либо посредством календаря. При необходимости заполнять либо изменять содержимое значение текстового поля «*Способ решения*».

Сохранение сделанных изменений производится нажатием кнопки «*Сохранить*». Если после работы с формой карточки запроса сохранение не требуется, то её нужно просто закрыть путем нажатия на кнопку «*Закрыть*». Кнопка «*Сброс*» нажимается при необходимости сбросить все значения формы к исходным, состояниям, до внесения изменений.

Для того чтобы удалить, при необходимости, запись о выбранном запросе на обслуживание из реестра запросов, необходимо нажать кнопку «*Удалить*», при подтверждении намерений из базы данных будет удалена запись о выбранном запросе, прикрепленные к записи файлы и история изменений. Кнопка «*Удалить*» доступна только для тех пользователей системы, у которых имеется соответствующая привилегия.

При работе незарегистрированного пользователя с системой, для завершения работы необходимо и достаточно воспользоваться стандартной системной кнопкой закрытия окна браузера, которая расположена в правом верхнем углу, в заголовке окна браузера «**[**x**]**».

Для зарегистрированных пользователей прошедших аутентификацию в системе, необходимо сначала завершить сеанс, путем нажатия кнопки «*Выйти*» (рисунок 12), подтвердить свое намерение завершить работу, в открывшемся диалоге, нажав кнопку «*ОК*». После чего воспользоваться стандартной системной кнопкой закрытия окна браузера «**[**x**]**», которая расположена в правом верхнем углу, в заголовке окна браузера.

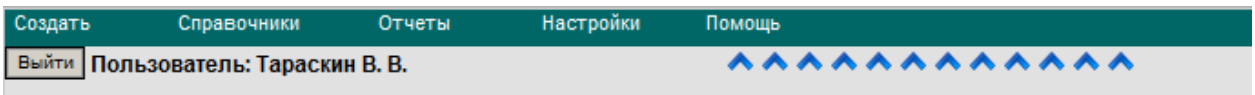

Рисунок 12 - Кнопка "Выйти"

2.5 Тестирование программного продукта

Тестирование – процесс выполнения прикладных программ с целью выявления ошибок.

Тестирование программного продукта проводилось несколькими способами. Было проведено полное тестирование правильного выполнения всех транзакций программы.

Следующим этапом тестирования являлось тестирование переходов между окнами диалогов. На этом этапе была проверенна правильность связей частей программы между собой. Были исследованы все возможные переходы из одних диалоговых окон в другие.

Программный продукт имеет сложную структуру. В связи с этим тестирование производилось поэтапно в процессе создания. Каждый модуль тестировался по отдельности. Далее модули объединялись и повторно проходили тестирование. Это делалось для проверки логических связей. Выявленные на стадии обработки ошибки исправлялись.

Отдельно проводилось тестирование всего программного продукта после завершения программирования. Для этих целей в качестве оператора (пользователя) привлекали несколько человек, не имеющих отношение к программированию. На этом этапе проверялась понятность программы для неопытного пользователя.

В процессе тестирования использовались следующие методы:

Тестирование по методу «*черного ящика*».

При тестировании по методу «черного ящика», программа не дает никаких сбоев. К данному тестированию можно отнести:

- не введение значений ключевых полей;
- различные переключения между режимами работы программы;
- ввод символов другого формата в поле таблицы.

Во всех выше описанных ситуациях программа ведет себя достаточно корректно, в некоторых случаях выдает сообщения об ошибке.

Тестирование по методу «*белого ящика*».

В данном методе тестирования многократно проверялся исходный код программы и работа некоторых узлов.

Во всех выше описанных ситуациях программа ведет себя достаточно корректно, в некоторых случаях выдает сообщения об ошибке.

ЗАКЛЮЧЕНИЕ

Цель дипломного проекта состояла в проектировании и реализации автоматизированной системы обеспечения работы сервисного центра. В ходе выполнения работы были получены следующие результаты:

- выполнено исследование предметной области;
- произведен обзор существующего программного обеспечения, подходящего для решения поставленной задачи автоматизации;
- выбраны инструментальные средства разработки программного продукта;
- выполнено проектирование базы данных;
- разработана и протестирована автоматизированная система обеспечения работы сервисного центра.

В дипломном проекте представлен один из множества подходов к разработке программного обеспечения информационного плана, дающий возможность на ранних этапах разработки учесть все нюансы будущего программного продукта, необходимый набор его функций, состав и структуру базы данных, что в дальнейшем исключает необходимость переработки уже написанных компонентов программы.

Система разработана с целью экономии времени при выполнении различных операций с компьютерным оборудованием, например, перемещение, изменение комплектующих, установка новых рабочих станций и другие. Все данные об оборудовании, зарегистрированном в данном программном продукте, сохраняются в базе данных. В любой момент их можно извлечь для печати необходимого документа, также изменить данные, тем самым поддерживая достоверность информации. Благодаря распределенной базе данных, информация базы может быть доступна на различных персональных компьютерах, подключенных к сети.

СПИСОК ИСПОЛЬЗОВАННЫХ ИСТОЧНИКОВ

1. ГОСТ 34.602–89. Техническое задание на создание автоматизированной системы [Текст] – Введен с 01.01.90.

2. ГОСТ 34.601–90. Автоматизированные системы. Стадии создания [Текст] – Введен с 01.01.92.

3. ГОСТ 34.201–89. Виды. Комплектность и обозначение документов при создании автоматизированных систем [Текст] – Дата введения с 01.01.90.

4. РД 50–34.698–90. Автоматизированные системы. Требования к содержанию документов [Текст] – Введен с 01.01.92.

5. Буч Г. Язык UML. Руководство пользователя. / Буч Г., Рамбо Дж., Джекобсон А. – Москва.: ДМК ПРЕСС, 2004. – 429 с.

6. Дубаков М.А. Веб-мастеринг средствами CSS. / М.А. Дубаков. – Санкт-Петербург.: БХВ-Петербург, 2002. – 544 с.

7. Зандстра М. PHP: объекты, шаблоны и методики программирования, 3-е издание. / Зандстра М. – Москва.: «Вильямс», 2010. – С. 560.

8. Кузин С.Г. Алгоритмы и методы предварительной обработки изображений. Методическое руководство лабораторного практикума. / С.Г. Кузин, А.Ф. Смирнов. – Нижний Новгород.: ННГУ, 2001. – 154стр.

9. Коржинский С.Н. Настольная книга Web-мастера: эффективное применение HTML, CSS и Javascript. / С.Н. Коржинский. – Москва.: Издательский торговый дом "Кнорус", 2000. – 320с.

10. Кригель А. SQL. Библия пользователя./ А. Кригель, Б. Трухнов. – Москва.: Издательство «Диалектика», 2002. – 548 стр.

11. Ларри Ульман. Ульман Л: Основы программирования на PHP. /Ларри Ульман. – Москва.: ДМК Пресс, 2001. – 288 с.

12. Нанда, А. Oracle PL/SQL для администраторов баз данных. / А. Нанда. – Москва.: Символ, 2008. – 496 с.

13. Суэринг С. PHP и MySQL. Библия программиста, 2-е издание. / Суэринг С., Конверс Т., Джойс П. – Москва.: «Диалектика», 2010. – 912 с.

14. Стивен Хольцнер . PHP в примерах. / Стивен Хольцнер. – Москва.: 000 «Бином-Пресс», 2007. – 352 с

15. Титтел Э. HTML, XHTML и CSS для чайников, 7-е издание. / Титтел Э., Ноубл Дж. – Москва.: «Диалектика», 2011. – 400 с.

16. Фейерштейн С. Oracle PL/SQL для профессионалов / С. Фейерштейн, Б. Прибыл. – Санкт-Петербург.: Питер, 2005. – 941 с.