

МИНИСТЕРСТВО НАУКИ И ВЫСШЕГО ОБРАЗОВАНИЯ РОССИЙСКОЙ ФЕДЕРАЦИИ  
федеральное государственное бюджетное образовательное учреждение  
высшего образования  
«Тольяттинский государственный университет»

Институт математики, физики и информационных технологий  
(наименование института полностью)

---

Кафедра «Прикладная математика и информатика»  
(наименование)

09.03.03 Прикладная информатика  
(код и наименование направления подготовки, специальности)

---

Бизнес-информатика  
(направленность (профиль) / специализация)

---

## **ВЫПУСКНАЯ КВАЛИФИКАЦИОННАЯ РАБОТА (БАКАЛАВРСКАЯ РАБОТА)**

на тему «Разработка web-приложения для учета заявок на техническую поддержку»

Обучающийся

А.И. Бедняков

(Инициалы Фамилия)

(личная подпись)

Руководитель

к.т.н., доцент, О.В. Аникина

ученая степень (при наличии), ученое звание (при наличии), Инициалы Фамилия)

Тольятти 2022

## Аннотация

Бакалаврская работа, 78 страниц, 31 рисунок, 25 таблиц, 35 источников, 1 приложение.

КОМПЬЮТЕР, ПРОГРАММНЫЙ МОДУЛЬ, ЗАЯВКИ, ТЕХПОДДЕРЖКА, ИНФОРМАЦИОННАЯ СИСТЕМА, МОДЕЛИРОВАНИЕ, РАЗРАБОТКА, БАЗА ДАННЫХ, ДАННЫЕ, WEB-ИНТЕРФЕЙС

Объектом исследования является государственное автономное учреждение Иркутской области «Экспертиза в строительстве Иркутской области», сокращенное наименование – ГАУИО «Ирэкспертиза».

Предметом исследования является автоматизация техподдержки пользователей ГАУИО «Ирэкспертиза».

Цель работы – повышение эффективности работы Отдела информационного обеспечения путем разработка web-приложения для учета заявок на техническую поддержку.

Результатом работы является web-приложение для учета заявок на техническую поддержку.

Область применения – ГАУИО «Ирэкспертиза».

Программный продукт реализован в PHP 5.2.12, HTML5 и хорошо подходит для работы с базой данных, созданной в СУБД MySQL.

Пояснительная записка к бакалаврской работе выполнена в текстовом редакторе Microsoft Office Word 2016 и представлена в электронной версии в соответствующем электронном курсе.

## Оглавление

Введение.....	4
Глава 1 Обзор предметной области.....	7
1.1 Описание организации .....	7
1.2 Характеристика основной деятельности отдела информационного обеспечения .....	10
1.3 Определение требований к web-приложению для учета заявок на техническую поддержку.....	13
1.4 Анализ существующих разработок для автоматизации техподдержки	16
1.5 Выбор и описание средств проектирования и разработки к web-приложения.....	21
1.6 Функциональное моделирование учета выполнения заявок в условиях автоматизации .....	33
Глава 2 Проектирование web-приложения для учета заявок на техническую поддержку .....	36
2.1 Описание функционального назначения системы посредством диаграммы вариантов использования.....	36
2.2 Проектирование базы данных.....	38
2.3 Информационное и техническое обеспечение web-приложения.....	45
2.4 Построение архитектуры программного модуля техподдержки .....	48
2.5 Разработка серверной части модуля техподдержки.....	52
2.6 Разработка клиентской части модуля техподдержки.....	54
Глава 3 Обоснование экономической эффективности разработки и внедрения web-приложения .....	60
3.1 Выбор и обоснование методики расчёта экономической эффективности разработки и внедрения web-приложения.....	60
3.2 Расчёт показателей экономической эффективности .....	62
Заключение .....	70
Список используемой литературы и используемых источников.....	72
Приложение А Экранные формы .....	76

## Введение

Актуальность темы исследования заключается в том, что автоматизация любого процесса предполагает использование технических и программных средств. Однако наполняемость компьютерной и офисной техникой, программным обеспечением требует и соответствующего решения вопросов, связанных со сбоями в их работе. Поэтому как никогда актуальной является своевременное выявление таких сбоев, быстрая ответная реакция на них и устранение. Для своевременности действий необходимо организовать службу технической поддержки с системой управления заявками и инцидентами.

Правильно организованная система управления заявками и инцидентами может уменьшить время, необходимое, чтобы найти необходимые решения конфликтным ситуациям, как на программном, так и на аппаратном уровне, повышает точность и своевременность информации, устраняет избыточность.

Современные системы управления заявками и инцидентами являются одним из соответствующих инструментов таких автоматизаций. Одним из важных аспектов этого процесса является необходимость ускорить время выхода на рынок новых, сосредоточенных на определенной аудитории, продуктов.

Информационные технологии используются в разных областях человеческой деятельности. Это требует осознанного применения теоретических положений и инструментальных средств разработки баз данных и информационных систем.

Учитывая наличие современных инструментов ускоренного развития решений (дизайнеров форм, образцов для создания, процессов и т.д.), современные системы управления заявками и инцидентами стали одним из инструментов, с помощью которого можно существенно ускорить процесс внедрения и снизить стоимость разработки в качестве средства автоматизации, на самом деле, доносящих продукт до потребителя.

Объектом исследования является государственное автономное учреждение Иркутской области «Экспертиза в строительстве Иркутской области», сокращенное наименование – ГАУИО «Ирэкспертиза».

Предметом исследования является автоматизация техподдержки пользователей ГАУИО «Ирэкспертиза».

Целью бакалаврской работы является повышение эффективности работы Отдела информационного обеспечения путем разработка web-приложения для учета заявок на техническую поддержку.

Для достижения целей были сформулированы следующие задачи [1]:

- дать характеристику предметной области;
- изучить бизнес-процессы техподдержки пользователей и определить требования к web-приложению техподдержки пользователей;
- рассмотреть существующие системы технической поддержки пользователей;
- выполнить выбор средств разработки web-приложения техподдержки пользователей;
- разработать концептуальную модель данных предметной области и построить модель структуры базы данных web-приложения техподдержки пользователей;
- разработать архитектуру web-приложения техподдержки пользователей и выполнить описание его серверной и клиентской части;
- выполнить разработку интерфейса web-приложения техподдержки пользователей;
- оценка эффективности внедрения разработанного программного продукта.

Выпускная квалификационная работа состоит из введения, трех глав, заключения, списка использованных источников и приложений.

В первой главе выполняется описание предметной области, рассматривается организационная структура компании. Также в этой главе

выполняется моделирование бизнес-процессов работы Отдела информационного обеспечения по приему заявок пользователей.

Также в этой главе выполняется постановка задачи автоматизации, обзор программных аналогов и проводится обоснование проектных решений.

Во второй главе выполняется проектирование и разработка web-приложения техподдержки пользователей.

В данной главе выполняется проектирование системы, рассматривается информационное обеспечение задачи, выполняется моделирование базы данных.

Далее во второй главе описывается процесс проектирования информационного программного обеспечения задачи, представляются экранные формы, демонстрирующие работу системы и их описание.

Третья глава выпускной квалификационной работы посвящена расчетам и обоснованию экономической эффективности проекта.

Выбирается методика расчета экономической эффективности, согласно выбранной методики рассчитываются экономические показатели, выполняется их анализ.

Результатом исследования является разработанное web-приложение техподдержки пользователей ГАУИО «Ирэкспертиза».

# Глава 1 Обзор предметной области

## 1.1 Описание организации

Объектом исследования в работе является государственное автономное учреждение Иркутской области «Экспертиза в строительстве Иркутской области», сокращенное наименование – ГАУИО «Ирэкспертиза» [17].

Организация была создана в соответствии с постановлением Правительства Иркутской области от 6 сентября 2012 г. № 413-рп «О создании государственного автономного учреждения Иркутской области «Экспертиза в строительстве Иркутской области».

Основными задачами деятельности ГАУИО «Ирэкспертиза» являются:

- проведение государственной экспертизы проектной документации и государственной экспертизы результатов инженерных изысканий на территории Иркутской области;
- проверка достоверности определения сметной стоимости инвестиционных проектов, строительства, реконструкции и (или) технического перевооружения (если такое перевооружение связано со строительством или реконструкцией), осуществляемых с привлечением средств федерального и государственного бюджетов.

Министерством строительства и дорожного хозяйства Иркутской области:

- осуществлены функции и полномочия учредителя создаваемого учреждения;
- назначен руководитель учреждения и с ним заключен трудовой договор;
- Устав учреждения разработан и утвержден по согласованию с Министерством имущественных отношений Иркутской области.

Организационная структура управления предприятием – это совокупность взаимоувязанных элементов, которые функционируют и развиваются как единственное целое.

Приказом от 02.03.2022 года № 39-упр «Об утверждении Штатного расписания государственного автономного учреждения Иркутской области «Экспертиза в строительстве Иркутской области» утверждено новое Штатное расписание с 01.03.2022 по 31.12.2022 года:

Структура содержит в себе следующие отделы:

- Администрация – 3 единицы;
- Отдел бухгалтерского, налогового учета и отчетности – 4 единицы;
- Отдел кадровой и административно-хозяйственной работы – 6 единиц;
- Отдел правового сопровождения – 8 единиц;
- Отдел экспертизы объектов инженерного обеспечения – 8 единиц;
- Отдел экологической безопасности и организация строительства – 6 единиц;
- Отдел общестроительных решений – 11 единиц;
- Отдел инженерных изысканий – 7 единиц;
- Отдел по проверке достоверности сметной стоимости линейных объектов и специальных работ – 9 единиц;
- Отдел по проверке достоверности сметной стоимости объектов социальной сферы – 11 единиц;
- Отдел по проверке достоверности сметной стоимости объектов коммунальной инфраструктуры и специальных работ – 11 единиц;
- Отдел по проверке достоверности сметной стоимости проектно-изыскательских работ – 4 единицы.
- Отдел информационного обеспечения – 4 единицы.

В управлении данной организацией используется линейно функциональная структура, которая представлена на рисунке 1.



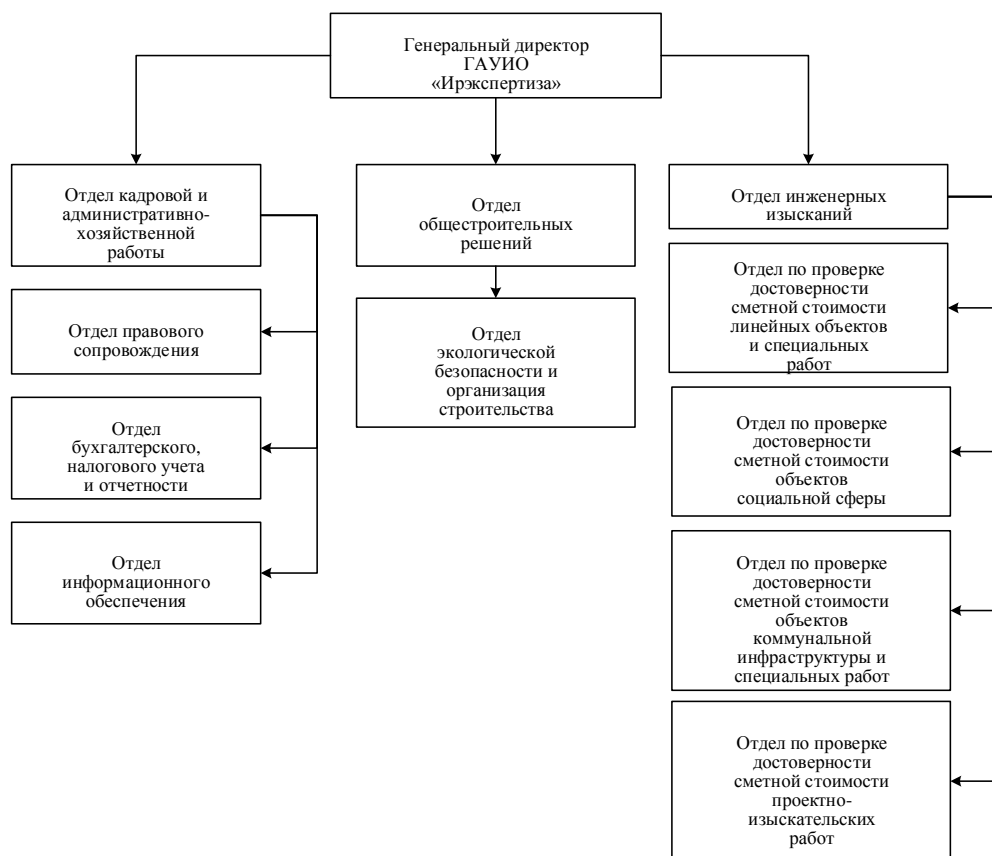


Рисунок 1 – Структурная схема ГАУИО «Ирэкспертиза»

Система руководства построена по линейно-иерархическому принципу. Генеральный директор осуществляет текущее общее руководство организацией. Каждый отдел имеет руководителя и сотрудников согласно штатного расписания.

Также в организации имеется отдел информационного обеспечения, который обеспечивает бесперебойную работы ИТ-структуры организации и осуществляет информационную и техническую поддержку пользователей.

Деятельность организации регулируется нормативными актами, утверждаемыми руководством организации, законами РФ.

Отдел информационного обеспечения ГАУИО «Ирэкспертиза» был создан с целью выполнения заявок на ремонт и обслуживание компьютеров и оргтехники, а также для устранения проблем в работе программного обеспечения.

Организационная структура отдела информационного обеспечения представлена на рисунке 2.



Рисунок 2 – Организационная структура отдела ИО

Сотрудники отдела выполняют разработку и сопровождения программных продуктов, используемых в организации; руководители осуществляют контроль за работой сотрудников служб отдела.

## **1.2 Характеристика основной деятельности отдела информационного обеспечения**

Основной бизнес процесс, который рассматривается в данной работе – «Учет заявок на техническую поддержку в Отделе информационного обеспечения ГАУИО «Ирэкспертиза».

Входными потоками в рассматриваемой деятельности являются заявки пользователей, журнал учета техники и журнал учета программного обеспечения.

Деятельность отдела информационного обеспечения регулируется действующим законодательством РФ, должностными инструкциями сотрудников, правилами работы с заявками.

Исполнителями являются сотрудники отдела информационного обеспечения, которые используют пакет офисных программ.

Выходными потоками являются фиксация выполнения заявки в журнале и отчетная документация [24] (рисунок 3).

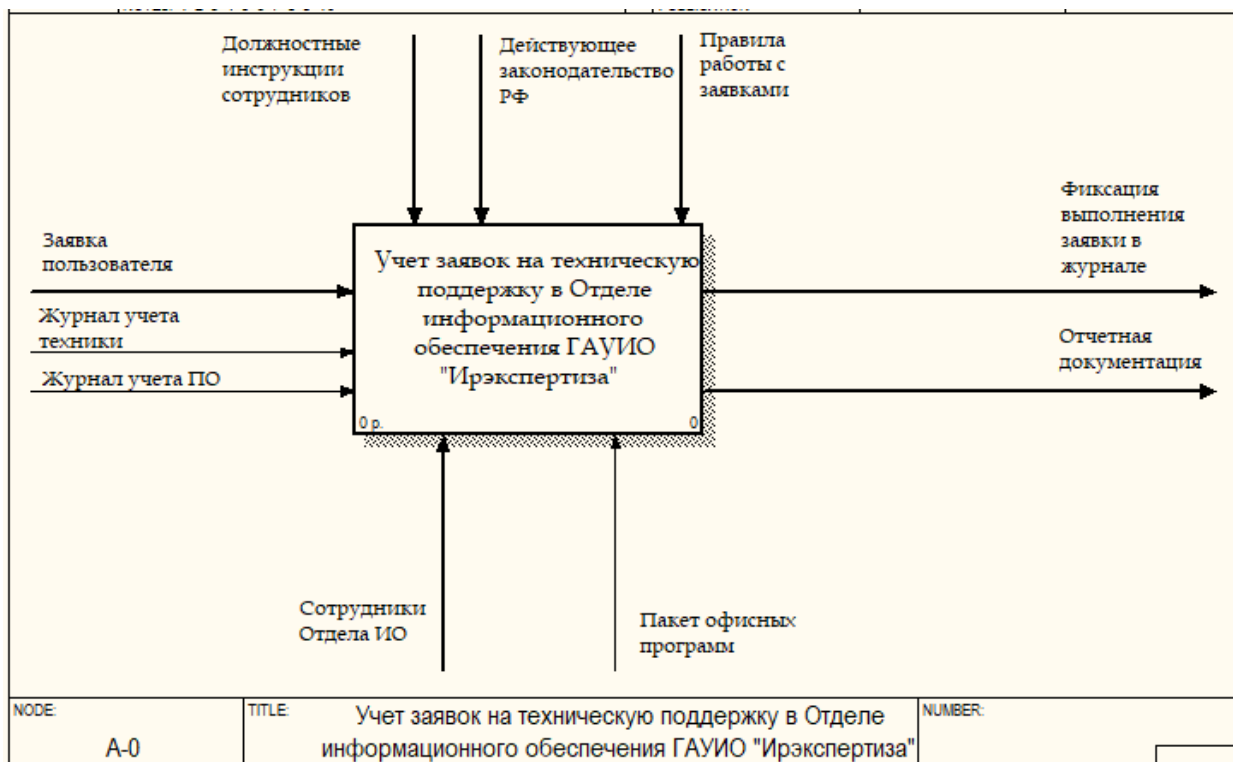


Рисунок 3 – Контекстная диаграмма бизнес-процесса «Как-есть»

Декомпозиция функциональной диаграммы выполнена на рисунке 4.

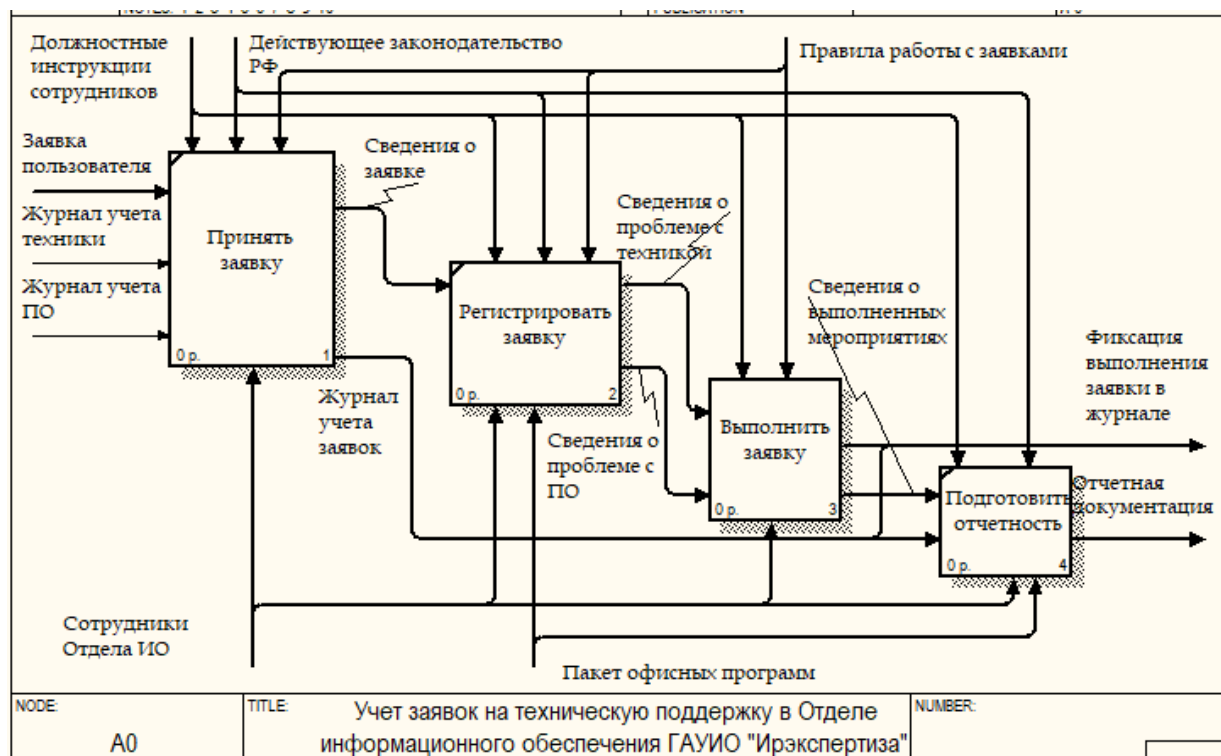


Рисунок 4 – Диаграмма декомпозиции бизнес-процесса «Как-есть»

Бизнес-процесс «Учет заявок на техническую поддержку в Отделе информационного обеспечения ГАУИО «Ирэкспертиза» разбивается на четыре работы:

- Принять заявку.
- Регистрировать заявку
- Выполнить заявку.
- Подготовить отчетность.

Рассмотрим процесс выполнения заявки более подробно (рисунок 5).

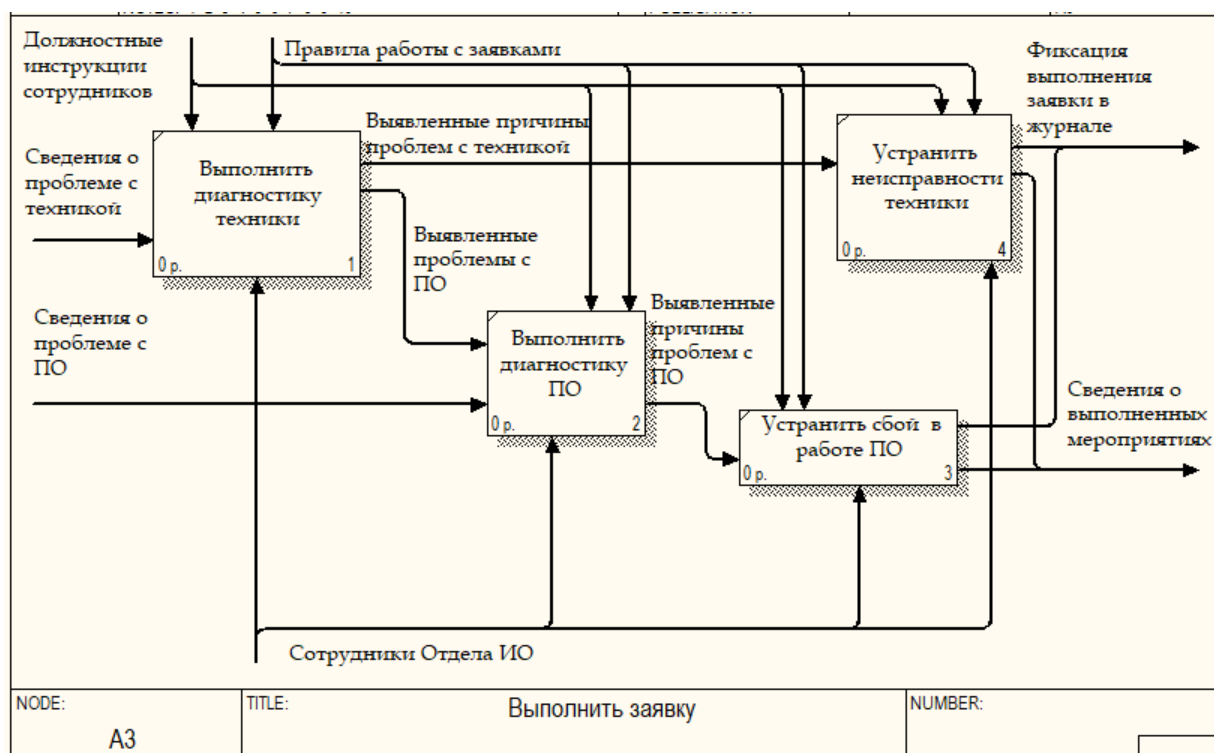


Рисунок 5 – Диаграмма декомпозиции работы «Выполнить заявку»

Работа «Выполнить заявку» разбивается на четыре компонента:

- Выполнить диагностику техники.
- Выполнить диагностику ПО.
- Устранить неисправности техники.
- Устранить сбои в работе ПО.

Проблемным местом для деятельности отдела информационного обеспечения является прием и своевременное обслуживание заявок

пользователей.

В настоящее время данный процесс организован следующим образом. Пользователь, используя телефонную связь, подает заявку в устной форме сотруднику отдела информационного обеспечения, кратко описывая существующую неисправность или проблему.

Сотрудник заносит ее в специальный журнал, направляется к пользователю и устраняет неисправность. После этого сотрудник записывает сведения о выполненной работе, израсходованном времени и материалах в журнал учета.

После этого данный пользователь обязан продублировать отчет о заявке в специальном журнале, а также по истечении некоторого периода времени, составить отчет на имя руководителя отдела информационного обеспечения.

Такой процесс имеет следующие недостатки:

- большое количество времени, затрачиваемое на регистрацию заявок и запись о выполненных мероприятиях;
- возможность допущения ошибок в записях;
- большая трудоемкость для сотрудников отдела.

Кроме того, раз в месяц необходимо подготавливать отчет о замеченных неисправностях и неполадках, в целях планирования закупки расходных материалов и наблюдения за статистикой поломок. Для этого сотруднику отдела необходимо выписать и систематизировать все записи за месяц или другой необходимый период, что очень трудоемко.

Поставленная задача будет автоматизирована путем разработки web-приложения для учета заявок на техническую поддержку.

### **1.3 Определение требований к web-приложению для учета заявок на техническую поддержку**

Схема документооборота в процессе подачи заявки и ее выполнения представлена на рисунке 6 [21].

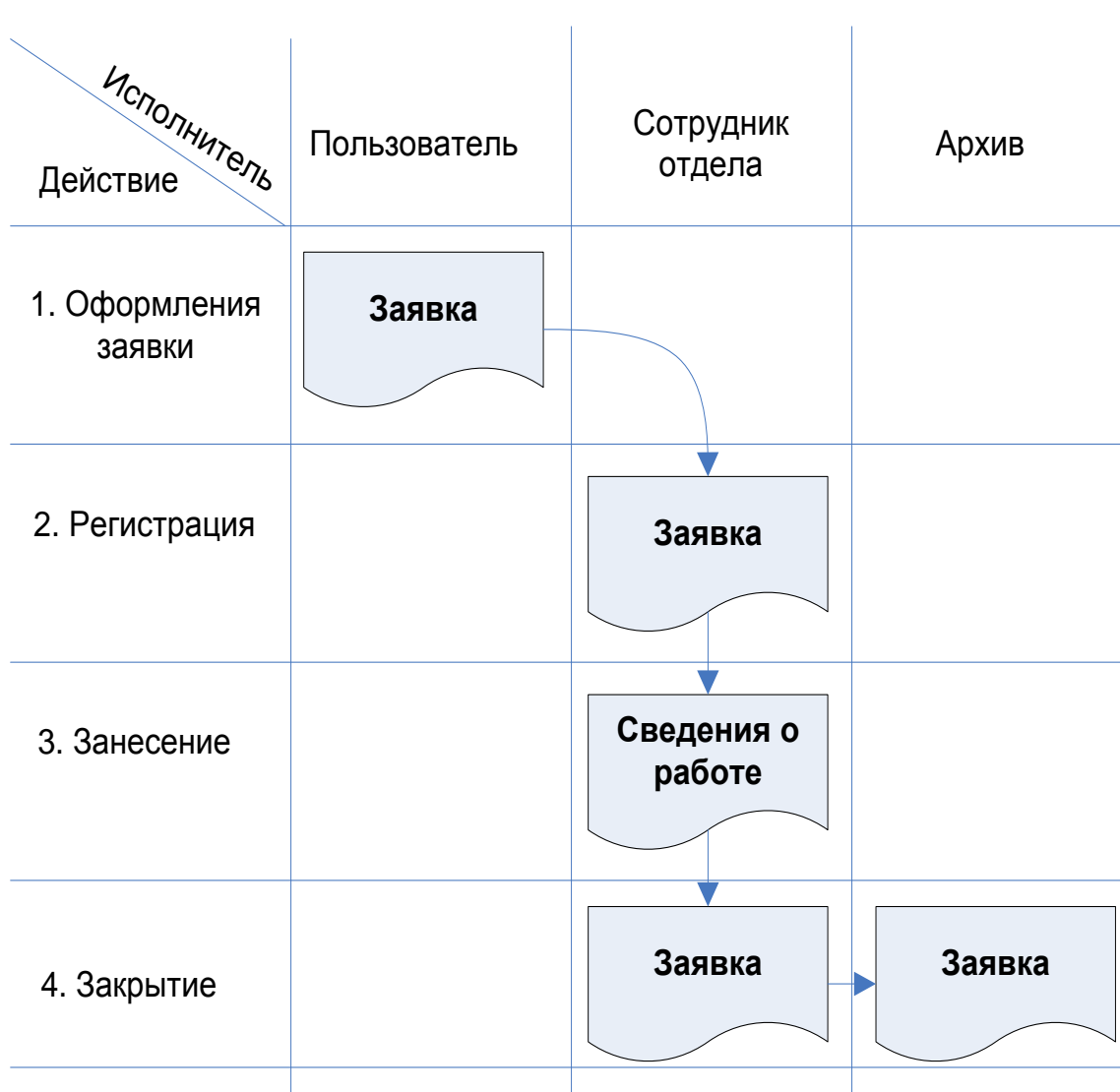


Рисунок 6 – Схема документооборота заявок о неисправностях

По этой схеме перед выполнением заявки и перемещением ее в архив, ее необходимо зарегистрировать, выполнить и закрыть с описанием выполненных работ.

Этот процесс будет автоматизирован созданием Web-ориентированного приложения с базой данных и Web-интерфейсом. При этом закрытие заявки происходит после внесения перечня работ, выполненных по ней [7].

Рассмотрим учет времени работы с заявками пользователей при существующей системе работы с заявками и в случае использования автоматизированной системы (таблица 1).

Таблица 1 – Учет времени работы с заявками пользователей при существующей системе работы с заявками и в случае использования автоматизированной системы

Действие	Всего действий (за 8 рабочих часов)	Затраты времени до автоматизации, мин.	Всего, мин.	Затраты времени после автоматизации, мин.	Всего, мин.
Принять заявку	5	20	100	2	10
Оформить заявку	5	30	100	10	50
Составить акт о проделанной работе	4	40	160	10	40
Итого:			360		100

Согласно расчетам, в итоге реализации процесса автоматизации планируется сократить затрачиваемое время с 360 минут (6 часов) до 100 минут ( $\approx 1,6$  часа). Дополнительная мотивация сотрудников предприятия за счет более удобного получения оформления заявок, а также более прозрачного управления ремонтом и устранением неисправностей ПК, сбоев программ и элементов ЛВС поднимет производительность их работы.

При эксплуатации разрабатываемого модуля техподдержки для обеспечения безопасности будут приняты меры по защите информации [28]. Характеристика разграничений прав пользователей приведена в таблице 2.

Таблица 2 – Разграничение прав пользователей

Группы пользователей	Вход в систему	Формирование заявки	Доступ к справочникам	Доступ к заявкам
Пользователи	Чтение, Ввод	Полный	Нет	Нет
Сотрудники отдела информационного обеспечения	Чтение, Ввод	Полный	Полный	Полный

Рассмотрим последовательность работы с системой.

Сотрудник отдела информационного обеспечения первоначально заполняет справочники «Должности», «Отделы», «Сотрудники» на основании штатного расписания.

Далее формируется справочник «Пользователи», а также справочник «Неисправности». В ходе формирования Справочника Пользователи каждый пользователь получает логин и пароль для доступа в систему. Пользуясь этими данными, пользователь получает доступ к странице подачи заявки, где, используя справочники неисправности, оформляет заявку, изменяя содержание таблицы «Заявки».

Оформленная заявка отображается у сотрудника отдела информационного обеспечения, который, выполнив заявку, вносит работы по ней, дату работ, после чего изменяется содержание таблицы «Заявки», заявка попадает в разряд архив.

#### **1.4 Анализ существующих разработок для автоматизации техподдержки**

Рассмотрим основные функции некоторых представителей программного обеспечения, позволяющих автоматизировать работу отделов, подробных отделу информационного обеспечения.

Система Web Help Desk компании SolarWinds [3].

Приложение обладает широкой функциональностью, неприхотливо в установке и обслуживании, а главное – позволяет эффективно контролировать процесс технической поддержки.

Поддерживаемые операционные системы: Mac OS X Server 10.4 или выше, Windows 2000 или выше, Red Hat Linux и производные, Debian Linux и производные, Sun Solaris 8 или выше.

Поддерживаемые базы данных: MySQL 5 или выше, MS SQL 2000 или выше, Oracle 8i или выше, OpenBase SQL, PostgreSQL 8 или выше, Frontbase 4 или выше.



Аппаратные требования: 768 Мбайт оперативной памяти, 1 Гбайт пространства на диске.

Интерфейс приложения представлен на рисунке 7.

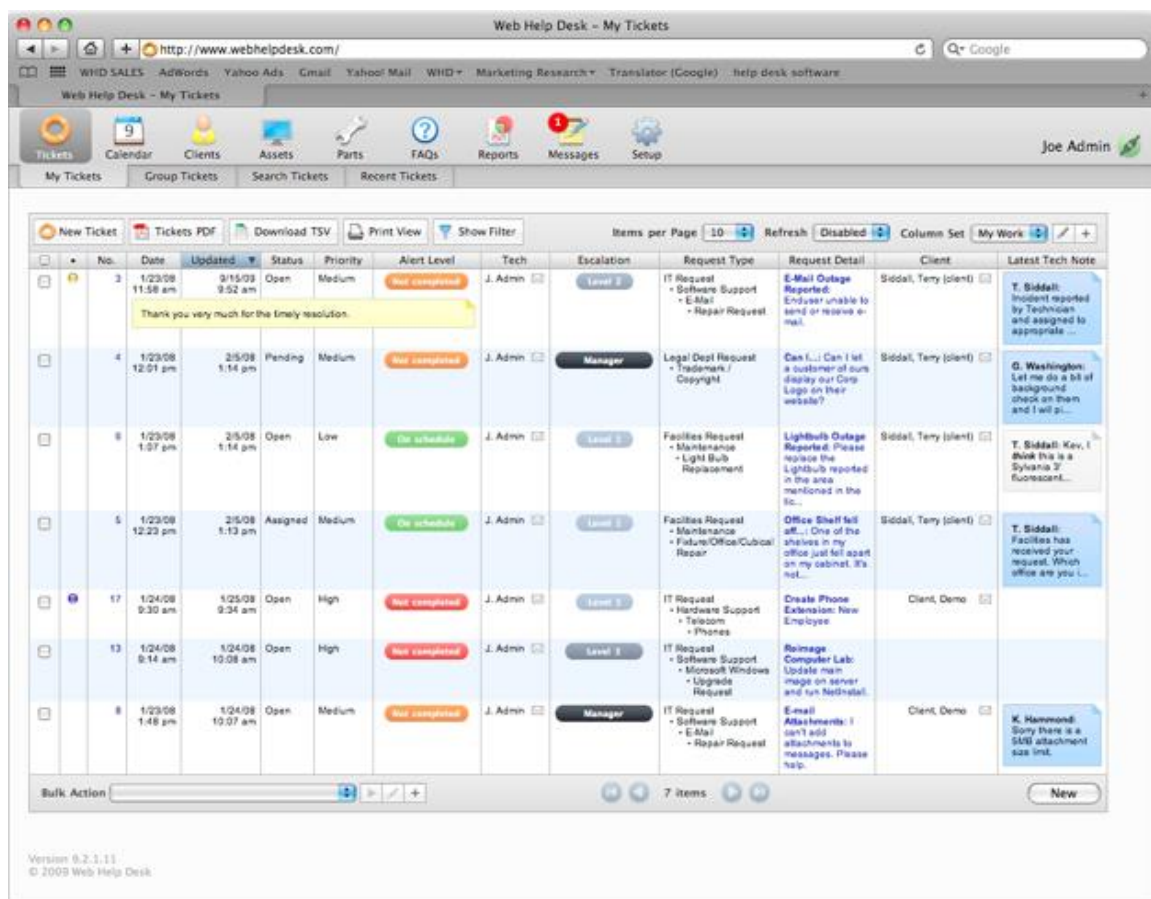


Рисунок 7 – Интерфейс системы Web Help Desk

Стоимость: бесплатная версия для одного сотрудника техподдержки, версия Lite Edition – 180-300 долларов на сотрудника техподдержки, версия Pro Edition – 480-900 долларов на сотрудника техподдержки. Для некоммерческих организаций действуют льготные цены.

Система Web Help Desk предназначена для отделов технической поддержки и консалтинговых агентств, которым требуется решение для отслеживания клиентских заявок, активов и связанных с ними обращений, а также для ведения календаря и просмотра отчетов.

Web Help Desk помогает управлять активами и обращениями клиентов.

Возможности:

- несколько видов отчетов по заявкам;
- простой и удобный веб-интерфейс;
- легкая процедура установки;
- автоматическое распределение заявок по сотрудникам;
- готовые формы заявок и инструменты самостоятельного поиска решений для клиентов;
- возможность отправки заявок по электронной почте;
- возможность ведения счетов и составления накладных;
- рассылка электронных уведомлений;
- мощный механизм поиска по клиентам, заявкам, активам, заказам на покупку и часто задаваемым вопросам.

Система Кларис ServiceDesk компании CLARIS [2].

Интерфейс системы представлен на рисунке 8.

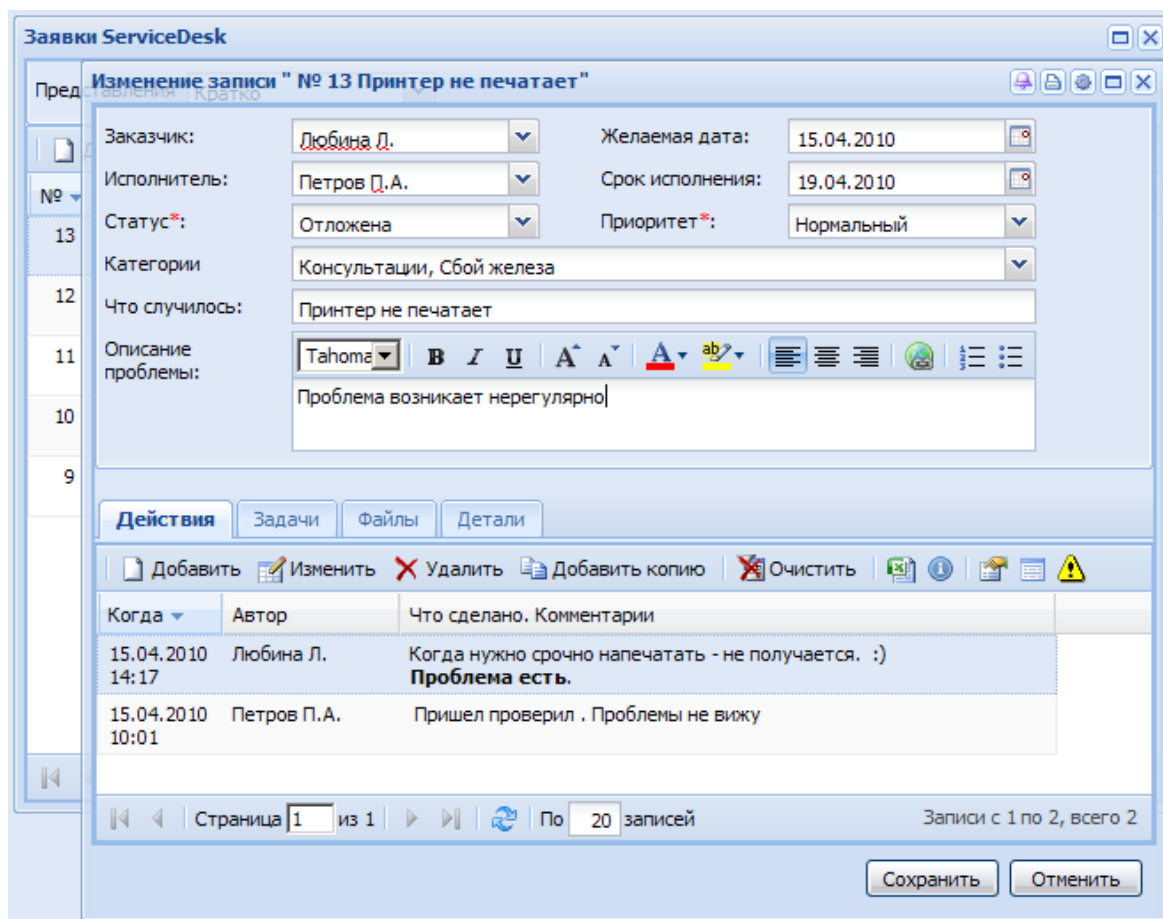


Рисунок 8 – Интерфейс системы Кларис ServiceDesk

Кларис Service Desk – web система для организации работы службы технической поддержки компаний с удобным и эргономичным интерфейсом.

Кларис Service Desk позволяет с минимальными затратами организовать работу с обращениями клиентов, не терять запросы, видеть состояния и историю выполнения заявок, контролировать сроки, анализировать показатели.

При использовании Кларис Service Desk процесс поддержки выходит на качественно новый уровень, особенно когда нужно поддерживать несколько различных по своему составу и наполнению программно-аппаратных комплексов.

Особенности Кларис ServiceDesk:

- удобный и понятный для пользователей интерфейс позволяет направлять запросы в службу поддержки;

- система доступна круглосуточно, с любого компьютера, имеющего выход в интернет. Количество сотрудников компании, количество филиалов и расположение офисов не имеет значения. Вы не привязаны больше к офису;

- стандартный способ регистрации заявок и выдачи заданий специалистам поддержки;

- права доступа к каждой заявке (полю заявки) определяются ролью пользователя и бизнес-правилами компании. Каждый видит только то, что необходимо для работы. Никто не видит лишнего.

- контроль за последовательностью исполнения работ, затраченным временем и ресурсами;

- оперативное оповещение задействованных сотрудников. E-mail и SMS;

- отчётность по затратам времени и средств на выполнение заявок

Система LiveAgent Help Desk компании Quality Unit [19].

Интерфейс системы представлен на рисунке 9.

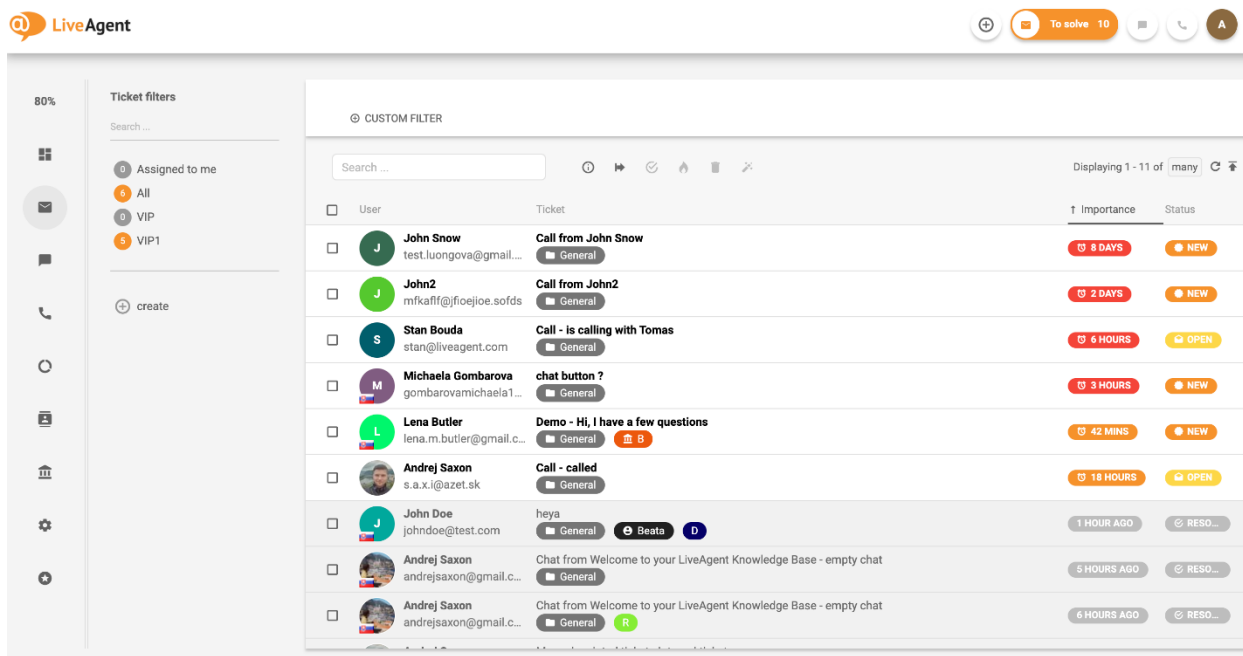


Рисунок 9 – Интерфейс системы LiveAgent

Программное обеспечение LiveAgent – это многоканальное программное обеспечение службы поддержки, которое упрощает общение с вашими клиентами посредством электронной почты, чатов, звонков и социальных сетей. Для лучшего управления LiveAgent преобразует всю коммуникацию в тикеты, которые можно назначать определенным агентам или группам - отделам. LiveAgent идеально впишется в вашу организационную структуру, даже если это компания с участием одного человека, предприятия малого и среднего бизнеса или компании среднего размера. В 2016 году более 15 000 компаний доверяют LiveAgent и используют его в качестве программного обеспечения для обслуживания клиентов.

Возможности программы:

- автоматический разбор email-сообщений;
- обработка сообщений с внешних и внутрикорпоративных почтовых ящиков;
- белый и черный лист адресов;
- ручной и автоматический разбор сообщений;
- настраиваемые правила разбора;

- формирование списка ответственных;
- возможность создания поручения в рамках инцидента;
- предложение решений проблемы;
- контроль решения инцидента;
- учет рабочего времени, затраченного на инцидент;
- файлы и комментарии;
- автоматическое информирование инициатора о статусе инцидента;
- настройка любых представлений инцидентов;
- группировка инцидентов по клиентам и контрагентам.

Все перечисленные выше программы являются платными, требуют дополнительных настроек и платного обслуживания. Поэтому необходимо создать собственную веб-ориентированную систему работы с заявками пользователей на базе СУБД, которая находится в свободном доступе.

### **1.5 Выбор и описание средств проектирования и разработки к web-приложения**

Для разработки модели данных используется программное средство Computer Associates ERWin. Данное программное обеспечение позволяет построить логическую и физическую модели, под выбранную СУБД и обладает удобным интерфейсом. Этот инструмент моделирования данных предлагает полный набор инструментов, который поддерживает практически весь жизненный цикл разработки базы данных [1].

AllFusion ERwin DM позволяет разработчикам баз данных немедленно приступить к работе, обеспечивая при этом среду для сохранения бизнес-правил, данных, метаданных, проектов баз данных, объектов баз данных и производственных стандартов.

AllFusion ERwin Data Modeler от CA может улучшить проектирование, создание и обслуживание высококачественных, высокопроизводительных баз данных, хранилищ данных и моделей корпоративных ресурсов данных. Этот выпуск включает в себя полностью переработанный механизм и позволяет

разработчикам баз данных документировать и упростить представление сложных проектов баз данных и структур данных, а также проводить инвентаризацию информационных активов и устанавливать корпоративные стандарты для управления данными.

Для разработки функциональной модели используется инструмент Computer Associates AllFusion Process Modeler. AllFusion Process Modeler – это мощное приложение для моделирования процессов, которое предлагает эффективное решение для моделирования, анализа, документирования и оптимизации бизнес-процессов. Он предоставляет все инструменты, необходимые для создания моделей различных бизнес-процессов и их анализа. Используя этот интеллектуальный инструмент, бизнес-профессионалы могут обнаруживать и документировать любые данные из любого места для обеспечения согласованности, ясности и повторного использования артефактов при крупномасштабной интеграции данных, управлении основными данными и управлении метаданными. Он может генерировать документацию и визуализировать работу различных схем. Кроме того, он также может получить доступ к истории и отсортировать все действия сеанса моделирования по объекту, типу объекта или типу транзакции, а также использовать параметры отмены и повтора. Он может автоматически генерировать модели данных и проекты баз данных для повышения эффективности и уменьшения количества ошибок. Это также позволяет пользователям улучшить аналитику данных и процесс принятия решений на предприятии, максимально расширяя возможности заинтересованных сторон использовать, понимать и доверять соответствующим данным.

Объектно-ориентированные модели проектирования строятся с использованием UML. UML расшифровывается как унифицированный язык моделирования и представляет собой стандартизированный язык визуального моделирования общего назначения в области разработки программного обеспечения. Он используется для определения, визуализации, построения и документирования основных артефактов программной системы. Это помогает

в разработке и характеристике, особенно тех программных систем, которые включают концепцию объектной ориентации. Он описывает работу как программных, так и аппаратных систем [22].

В рамках исследования будет разработано web-приложение «Техническая поддержка пользователей» средствами PHP и MySQL.

PHP (Hypertext Preprocessor) – широко используемый язык общего назначения сценариев с открытым исходным кодом [5].

PHP программист должен обеспечить средства для быстрого и эффективного достижения поставленных целей.

PHP обозначает язык сценариев на стороне сервера, что означает, что приложения, написанные на нем, работают на веб-серверах и не зависят от веб-браузера. Однако с годами область его использования изменилась, и в настоящее время язык программирования PHP входит в число лучших и самых популярных инструментов программирования для веб-разработки благодаря своим многочисленным достоинствам.

PHP в основном используется для веб-разработки, и он действительно преуспевает в этой области. Хотя изначально он использовался для создания динамических веб-страниц, разработчики предпочитают использовать этот язык сценариев для построения серверной части веб-приложений.

Доступно множество руководств PHP и других справочных материалов, которые облегчают разработку и могут помочь в сложных ситуациях. Изучение PHP сравнительно легко, хотя и не так просто, как, например, изучение Python, что еще больше увеличивает его популярность.

Использование PHP ускоряет загрузку страниц веб-сайта по сравнению со многими другими технологиями веб-разработки. В свою очередь, более низкое время загрузки является важным фактором ранжирования SEO, который способствует дальнейшему продвижению сайта, обеспечивая конкурентные преимущества. Более высокая скорость работы приложения обеспечивает удовлетворенность клиентов и, в сочетании с другими преимуществами, помогает формировать и сохранять клиентскую базу.

PHP позволяет подключаться практически к любому типу базы данных. Наиболее распространенным выбором является MySQL, главным образом потому, что он бесплатный, эффективный и популярный среди разработчиков.

PHP – это бесплатная технология, позволяющая значительно сократить бюджет разработки. Кроме того, большинство инструментов разработки, которые обычно используются в сочетании с PHP, являются программным обеспечением с открытым исходным кодом и могут использоваться бесплатно; таким образом, они дополнительно снижают стоимость проекта. Более того, существуют многочисленные фреймворки, такие как Laravel и CodeIgniter, и различные CMS, такие как Wordpress и Drupal, например, которые расширяют функциональность PHP и делают процесс разработки проще и эффективнее.

Самый распространенный сценарий работы PHP-сайта – стек LAMP. Это означает, что веб-сайт работает на веб-сервере Apache HTTP, развернутом в системе Linux, и использует MySQL в качестве базы данных. Все эти компоненты бесплатны, а стек хорошо протестирован, что подразумевает сокращение необходимых времени и средств на разработку.

Поскольку PHP встроенный язык, он является очень гибким по отношению к потребностям разработчика. PHP, как правило, рекомендуют для использования в сочетании с HTML

PHP предлагает встроенное программирование HTML, что является причиной невероятной синергии между этими двумя технологиями. В большинстве случаев PHP-скрипт не вмешивается в HTML-код веб-страницы, а вместо этого завершает его, оставаясь внутри границ, определенных тегами `<?php ?>`. И наоборот, если целая страница построена на PHP, можно интегрировать HTML-код с помощью скрипта. В этом случае отличием от традиционной HTML-страницы будет обратный порядок открывающих тегов: теги PHP будут предшествовать тегам HTML.



HTML (Hyper Text Markup Language) или язык гипертекстовой разметки – это язык разметки для Интернета, который определяет структуру веб-страниц [25].

Это один из самых основных строительных блоков каждого веб-сайта.

Поскольку HTML определяет разметку для конкретной веб-страницы, нужно, чтобы текст, изображения или другие встраивания отображались определенным образом.

Например, чтобы какой-то текст был большим, другой – маленьким, а какой-то – полужирным, курсивом или в виде маркированного списка.

В HTML есть «теги», которые позволяют это сделать. Итак, есть теги для создания заголовков, абзацев, слов, выделенных жирным шрифтом, слов, выделенных курсивом, и многого другого.

HTML-элемент состоит из открывающего тега, символа, содержимого и закрывающего тега. Некоторые элементы пусты, то есть у них нет закрывающего тега, а вместо этого есть источник или ссылка на контент, который вы хотите встроить на веб-страницу.

Примером пустого элемента является `<img>`, который используют для встраивания изображений на веб-страницу.

Теги HTML также принимают так называемые атрибуты. Эти атрибуты помещаются в открывающий тег и варьируются от стиля и идентификатора до классов. Они принимают значения, которые передают дополнительную информацию об элементе и помогают выполнять такие действия, как стилизация и манипулирование с помощью JavaScript.

HTML определяет структуру веб-страниц. Одной этой структуры недостаточно, чтобы веб-страница выглядела хорошо и интерактивно. Таким образом, необходимо использовать вспомогательные технологии, такие как CSS и JavaScript, чтобы сделать HTML-документ красивым и добавить интерактивности, соответственно [25].

Можно посмотреть на HTML, CSS и JavaScript таким образом: HTML – это структура, CSS – это внутренняя и внешняя отделка, а JavaScript – это функциональные особенности.

При сохранении созданного файла HTML необходимо использовать расширение .html.

CSS – это каскадные таблицы стилей. Их использование с HTML-страницами является передовой практикой разработки пользовательского интерфейса (UI), которая соответствует шаблону проектирования разделения задач [25].

CSS – это стандартный и предпочтительный механизм форматирования HTML- страниц.

В соответствии с шаблоном проектирования разделения задач и передовой практикой каскадные таблицы стилей обеспечивают центральное расположение, в котором информация о том, какие различные шрифты, цвета переднего плана, цвета фона, курсив и выделение должны применяться к различным элементам HTML на веб-странице.

Каскадные таблицы стилей также могут управлять размещением на странице различных частей страницы, таких как верхний и нижний колонтитулы, основной текст, содержимое статьи, разделы и отступления. Это чрезвычайно полезно, когда контент должен быть размещен совершенно по-разному в зависимости от того, просматривается ли он на настольном компьютере, планшете или смартфоне.

Каскадный характер файлов CSS объясняется тем, что информация о стиле для веб-страницы может быть определена в любом из трех разных мест, также известных как уровни стиля.

Предпочтительной практикой является размещение информации о стиле в отдельном файле с расширением .css.

Чтобы внести глобальное изменение в стиль сайта, просто нужно изменить файл стилей CSS, и все элементы на всех веб-страницах будут обновлены автоматически.

Улучшенные стили по сравнению с HTML – CSS имеет гораздо более широкий набор атрибутов, чем HTML.

Таблицы стилей позволяют оптимизировать содержимое для более чем одного типа устройств. Используя один и тот же HTML-документ, разные версии веб-сайта могут быть представлены для портативных устройств, таких как КПК и сотовые телефоны, или для печати.

Рассмотрим две технологии верстки макетов CSS Grid Layout и CSS Flexbox Layout: эти две технологии хорошо использовать вместе.

Существенное их отличие заключается в том, что Flexbox берет контент за основу, а Grid основан на самом макете и его общей верстке.

Flexbox - это одномерные макеты, а Grid - двумерные. Это означает, что если размещаются объекты в одном направлении, например, кнопка в заголовке, то необходимо использовать Flexbox.

Это даст больше гибкости, чем с помощью CSS Grid. Также будет легче работать с этим в будущем, и что наиболее важно, можно избежать ненужного написания кода.

Однако, если необходимо создать весь макет в двух измерениях, как со строкой, так и со столбцом, необходимо использовать CSS Grid.

В этом случае с помощью CSS Grid можно получить больше гибкости, и разметка станет проще, а код станет более понятным для дальнейшей работы.

Таким образом, при верстке макетов Web-страниц можно комбинировать оба метода: использовать Grid для всего макета, а затем Flexbox для выравнивания содержимого внутри заголовка. Таким образом, получается выгода от использования двух технологий верстки.

API занимают центральное место в сегодняшней обширной программной экосистеме. Существует практически бесконечное количество способов подключения различных веб-приложений, и API-интерфейсы поддерживают эту интеграцию.

Существуют разные категории API, основанные на разных протоколах, функциях и уровнях доступа.

Говоря об API, мы обычно говорим о подкатегории API, называемой веб-API. Веб-API – это API-интерфейсы, доступ к которым осуществляется с использованием протокола передачи гипертекста (HTTP), того же протокола, который используется для получения и отображения веб-страниц в браузерах.

Веб-API можно сгруппировать в четыре широкие категории:

Открытые API, к которым может получить доступ любой разработчик.

Партнерские API, доступ к которым имеют только авторизованные разработчики.

Внутренние API, доступ к которым имеют только внутренние команды.

Составные API, объединяющие несколько API.

Также существует три распространенных типа архитектуры API:

- REST – набор руководств по легковесным масштабируемым веб-API.
- SOAP – более строгий протокол для более безопасных API.
- RPC – протокол для вызова процессов, который может быть написан с помощью XML ( XML-RPC ) или JSON ( JSON-RPC ).

Создание многофункциональных приложений JavaScript, которые переносят возможности рабочего стола в Интернет, требует перемещения состояния с сервера на сторону клиента.

JavaScript (JS) – это кроссплатформенный объектно-ориентированный язык программирования, используемый разработчиками для создания интерактивных веб-страниц. Он позволяет разработчикам создавать динамически обновляемый контент, использовать анимацию, всплывающие меню, интерактивные кнопки, управлять мультимедиа и т. д. Использование JavaScript может быть как на стороне клиента, так и на стороне сервера. В то время как языки HTML и CSS используются для придания структуры и стиля веб-страницам, JavaScript используется для добавления интерактивных элементов, привлекающих внимание пользователей. Без JavaScript 90% веб-страниц в Интернете были бы статичными [12].

JavaScript широко используется для создания веб-сайтов и веб-приложений.

JavaScript – это язык сценариев, используемый для разработки веб-страниц. Разработанный в Netscape, JS позволяет разработчикам создавать динамические и интерактивные веб-страницы для взаимодействия с посетителями и выполнения сложных действий. Это также позволяет пользователям загружать содержимое в документ без перезагрузки всей страницы. Большинство веб-сайтов используют JavaScript для проверки и поддержки внешних приложений, включая PDF-документы, виджеты, флэш-приложения. Некоторые из крупнейших мировых технологических компаний используют JavaScript для улучшения взаимодействия с пользователем.

Различные фреймворки JavaScript используются для разработки и создания надежных веб-приложений. В таком приложении, как Google Maps, если пользователи хотят изучить карту, все, что им нужно сделать, это щелкнуть и перетащить мышью, чтобы получить подробный вид. Это работает на JavaScript, который взаимодействует с браузером без связи с серверами. Популярными внешними средами JavaScript, которые помогают создавать веб-приложения, являются React Native, React, Angular и Vue. Netflix и PayPal были разработаны с использованием фреймворка JavaScript AngularJS и интерфейсов прикладного программирования (API).

Задача долговременного хранения информации очень важна для сайта: хранение сообщений, дистанционное управление информацией на сайте и списка услуг центра.

Приложение PHP, которое использует базу данных для хранения информации (в частности, MySQL), всегда работает быстрее, чем приложения, созданные на основе файлов.

MySQL – одна из самых популярных и распространенных систем управления базами данных для Интернета, подходит для интернет-сайтов [18].

MySQL – это система, которая помогает эффективно хранить данные и управлять ими. База данных обычно хранит данные в структурированном

виде. Ее код написан на C и C++ и был протестирован с различными компиляторами на наличие ошибок и несоответствий.

Рассмотрим некоторые основные функции, связанными с MySQL:

Открытый исходный код

MySQL имеет открытый исходный код, что означает, что это программное обеспечение может быть загружено, использовано и модифицировано кем угодно. Он бесплатный и простой для понимания. Исходный код MySQL можно изучить и изменить в зависимости от требований. Он использует GPL, то есть лицензию GNU General Public, которая определяет правила и положения относительно того, что можно и что нельзя делать с помощью приложения.

Быстра и надежна

MySQL эффективно хранит данные в памяти, обеспечивая согласованность данных и отсутствие избыточности. Следовательно, доступ к данным и манипулирование ими с помощью MySQL выполняются быстро.

Масштабируемость

Масштабируемость относится к способности систем легко работать с небольшими объемами данных, большими объемами данных, кластерами машин и так далее. Сервер MySQL был разработан для работы с большими базами данных.

Типы данных

Он содержит несколько типов данных, таких как целые числа без знака, целые числа со знаком, число с плавающей запятой (FLOAT), двойное число (DOUBLE), символ (CHAR), переменный символ (VARCHAR), текст, большой двоичный объект, дата, время, дата-время, отметка времени, год и т. д. на.

Наборы символов

Он поддерживает различные наборы символов, в том числе latin1 (кодировка символов cp1252), немецкий, Ujis, другие наборы символов Unicode и так далее.

## Безопасна

Она обеспечивает безопасный интерфейс, поскольку имеет гибкую систему паролей и гарантирует, что они проверяются на основе хоста перед доступом к базе данных. Пароль шифруется при подключении к серверу.

## Поддержка больших баз данных

Она поставляется с поддержкой больших баз данных, которые могут содержать от 40 до 50 миллионов записей, от 150 000 до 200 000 таблиц и до 5 000 000 000 строк.

## Клиентские и вспомогательные программы

Сервер MySQL также поставляется со многими клиентскими и служебными программами. Сюда входят программы командной строки, такие как «mysqladmin», и графические программы, такие как «MySQL Workbench». Клиентские программы MySQL написаны на разных языках. Клиентская библиотека (код, инкапсулированный в модуль) может быть написана на C или C++ и будет доступна для клиентов с привязками C [34].

Язык структурированных запросов (SQL) – это стандартизированный язык программирования, который используется для управления реляционными базами данных и выполнения различных операций с данными в них [6]. SQL регулярно используется не только администраторами баз данных, но и разработчиками, пишущими сценарии интеграции данных, и аналитиками данных, которые хотят настроить и выполнить аналитические запросы.

SQL используется для следующих целей:

- изменение структуры таблиц и индексов базы данных;
- добавление, обновление и удаление строк данных;
- извлечение подмножеств информации из систем управления реляционными базами данных (RDBMS) – эта информация может использоваться для обработки транзакций, аналитических приложений и других приложений, которым требуется связь с реляционной базой данных.

SQL-запросы и другие операции принимают форму команд, написанных в виде инструкций, и объединяются в программы, которые позволяют пользователям добавлять, изменять или извлекать данные из таблиц базы данных [34].

Таблица является основной единицей базы данных и состоит из строк и столбцов данных. Одна таблица содержит записи, и каждая запись хранится в строке таблицы. Таблицы являются наиболее часто используемым типом объектов базы данных или структур, которые содержат или ссылаются на данные в реляционной базе данных. Другие типы объектов базы данных включают следующее:

Представления – это логические представления данных, собранных из одной или нескольких таблиц базы данных.

Индексы – это таблицы поиска, которые помогают ускорить функции поиска в базе данных.

В прикладных программах SQL используется как интерфейс прикладного программирования (API). Связка PHP и MySQL используется абсолютным большинством разработчиков динамических Web-приложений.

Поскольку для разработки сайта необходимо разработать динамический сайт, на котором будут использоваться PHP-скрипты, база данных, для отладки такого сайта нужен сервер.

Сервер в информационных технологиях – программный компонент вычислительной системы, который выполняет сервисные (обслуживающие) функции по запросу клиента, предоставляя ему доступ к определённым ресурсам или услугам.

Локальный сервер находится на компьютере пользователя.

Существует несколько видов готовых пакетов web-разработчиков. Ниже приведены самые распространённые:

- Denwer – Apache, MySQL, PHP;
- XAMPP – Apache, MySQL, PHP, Perl, FTP-server;
- JSAS – XAMPP + Joomla.



Отдадим предпочтение пакету Denwer. Пакет прост в установке и содержит все необходимые компоненты для работы с адресной книгой [10].

Денвер (от сокр. Д.н.в.р – джентльменский набор Web-разработчика) – набор дистрибутивов и программная оболочка, которые необходимы для создания и отладки сайтов на локальном ПК без подключения к сети Интернет под управлением ОС Windows.

Программное обеспечение, необходимое для создания сайта бесплатно и находится в свободном доступе. Скачать их можно с сайта разработчика Denwer.

При разработке данного сайта был установлен пакет Denwer версии 3 с компонентами (Apache 2, MySQL 5.0, PHP 5.2.4).

## **1.6 Функциональное моделирование учета выполнения заявок в условиях автоматизации**

Основной бизнес-процесс техподдержки пользователей – автоматизированный учет выполнения заявок на техническую поддержку в Отделе информационного обеспечения ГАУИО «Ирэкспертиза» представлен на контекстной диаграмме, выполненной в нотации IDEF0 на рисунке 10.

Входными потоками в рассматриваемой деятельности является Заявка пользователя, Информация о технике и информация о программном обеспечении.

Выходными потоками в рассматриваемой деятельности является отчетная документация, а также документация о выполнении заявки пользователя, сообщения об отклонении заявки и периодическая отчетность.

Деятельность отдела информационного обеспечения регулируется действующим законодательством РФ, должностными инструкциями сотрудников, правилами работы с заявками и инструкцией по работе с Web-приложением.

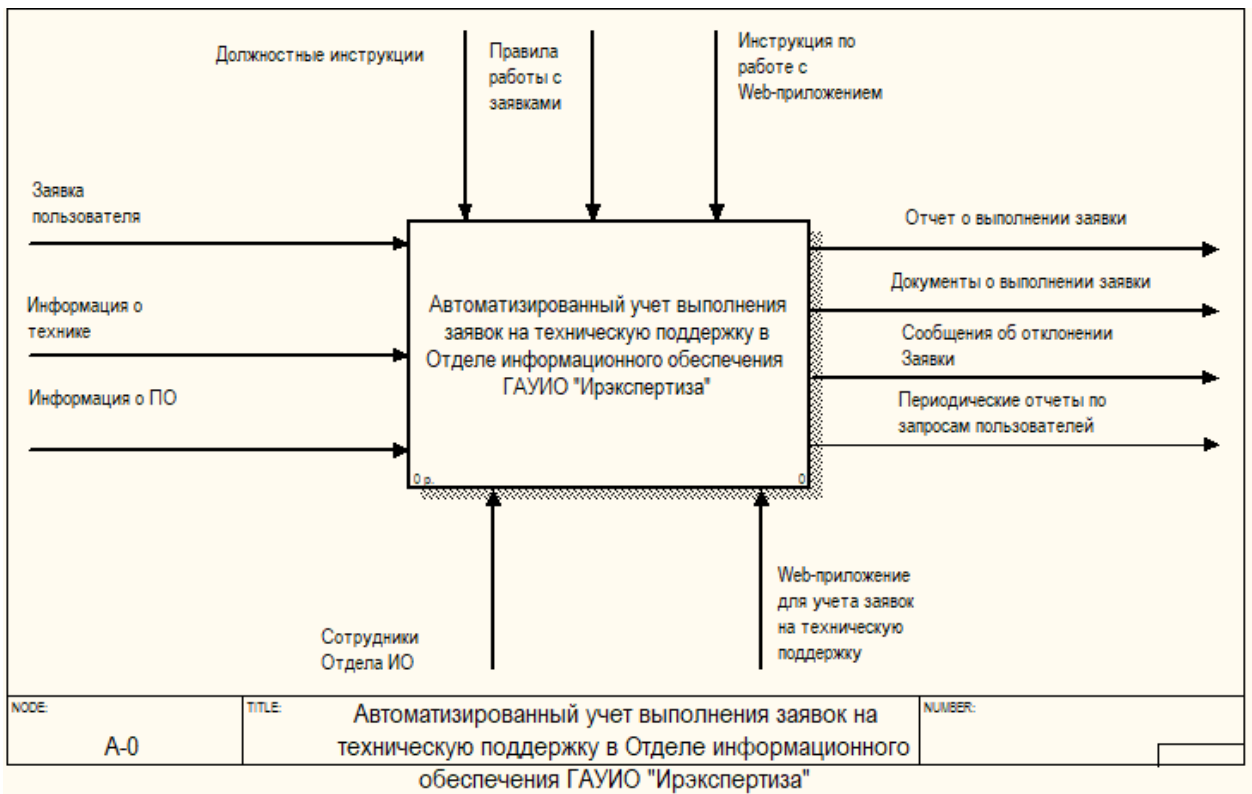


Рисунок 10 – Контекстная диаграмма бизнес-процесса «Как должно быть»

Декомпозиция данного процесса приведена на рисунке 11.

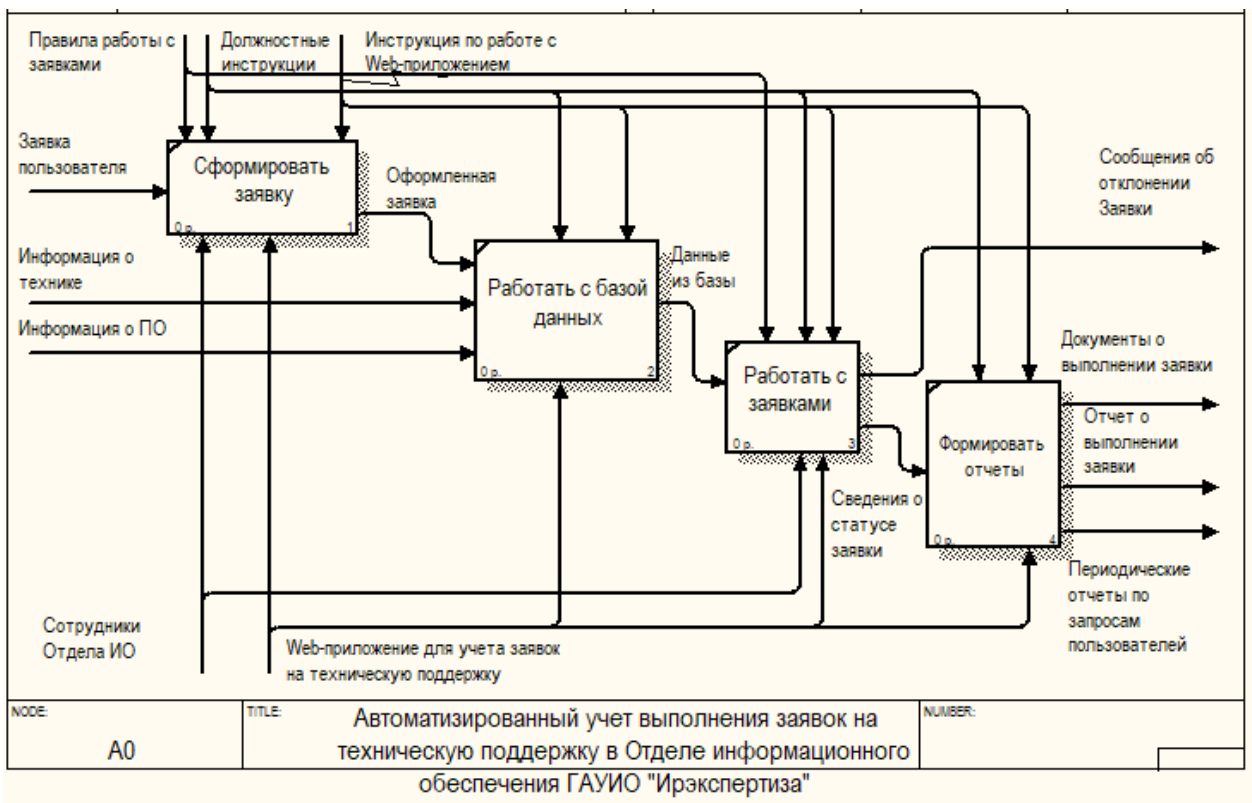


Рисунок 11 – Декомпозиция бизнес-процесса «Как должно быть»

Работу по обеспечению связи с клиентами, берет на себя программный модуль.

В условиях работы программного модуля данный процесс организован следующим образом. Пользователь, используя форму обратной связи или чат, подает заявку в отдел информационного обеспечения, кратко описывая существующую неисправность или проблему.

Программный модуль фиксирует заявку и посылает сигнал сотрудникам отдела ИО. Сотрудник отдела, ответственный за прием обращений пользователей, просматривает краткую характеристику проблемы и назначает исполнителя или отклоняет заявку, если данная проблема не решается в рамках техподдержки, с инструкциями пользователю для дальнейших действий.

После выполнения заявки, Сотрудник отдела, ответственный за прием обращений пользователей, заносит данные в базу и направляет пользователю документ об исполнении заявки.

По истечении некоторого периода времени, Сотрудник отдела, ответственный за прием обращений пользователей, составляет отчеты – плановые или по запросам, о состоянии выполнения заявок пользователей на имя руководителя отдела информационного обеспечения.

#### Выводы по 1 главе

В данной главе выполнено описание предметной области, рассмотрена организационная структура компании и Отдела информационного обеспечения. Также в этой главе выполнено моделирование бизнес-процессов работы Отдела информационного обеспечения по приему заявок пользователей «Как есть» и «Как должно быть».

Также в этой главе выполнена постановка задачи автоматизации, обзор программных аналогов и проведено обоснование проектных решений.

## Глава 2 Проектирование web-приложения для учета заявок на техническую поддержку

### 2.1 Описание функционального назначения системы посредством диаграммы вариантов использования

Диаграмма вариантов использования – это тип поведенческой диаграммы UML, который часто используется для анализа различных систем. Они позволяют визуализировать различные типы ролей в системе и то, как эти роли взаимодействуют с системой [23].

Диаграммы вариантов использования описывают высокоуровневые функции и область применения системы. Эти диаграммы также определяют взаимодействие между системой и ее действующими лицами.

Актеры – это внешние объекты, взаимодействующие с вашей системой. Это может быть человек, другая система или организация [31].

Выделим три роли для работы с web-приложением для учета заявок на техническую поддержку (таблица 3).

Таблица 3 – Роли пользователей web-приложения для учета заявок на техническую поддержку

Актер (роль)	Описание
Пользователь	Пользователь компьютерной техники, который подает заявку на техническую поддержку
Сотрудник	Сотрудник отдела информационного обеспечения (ИО), ответственный за прием обращений пользователей
Исполнитель	Сотрудник отдела ИО, ответственный за исполнение обращений пользователей

Рассмотрим диаграмму вариантов использования (прецедентов) работы web-приложения для выполнения заявок пользователей сотрудниками отдела ИО (рисунок 12).

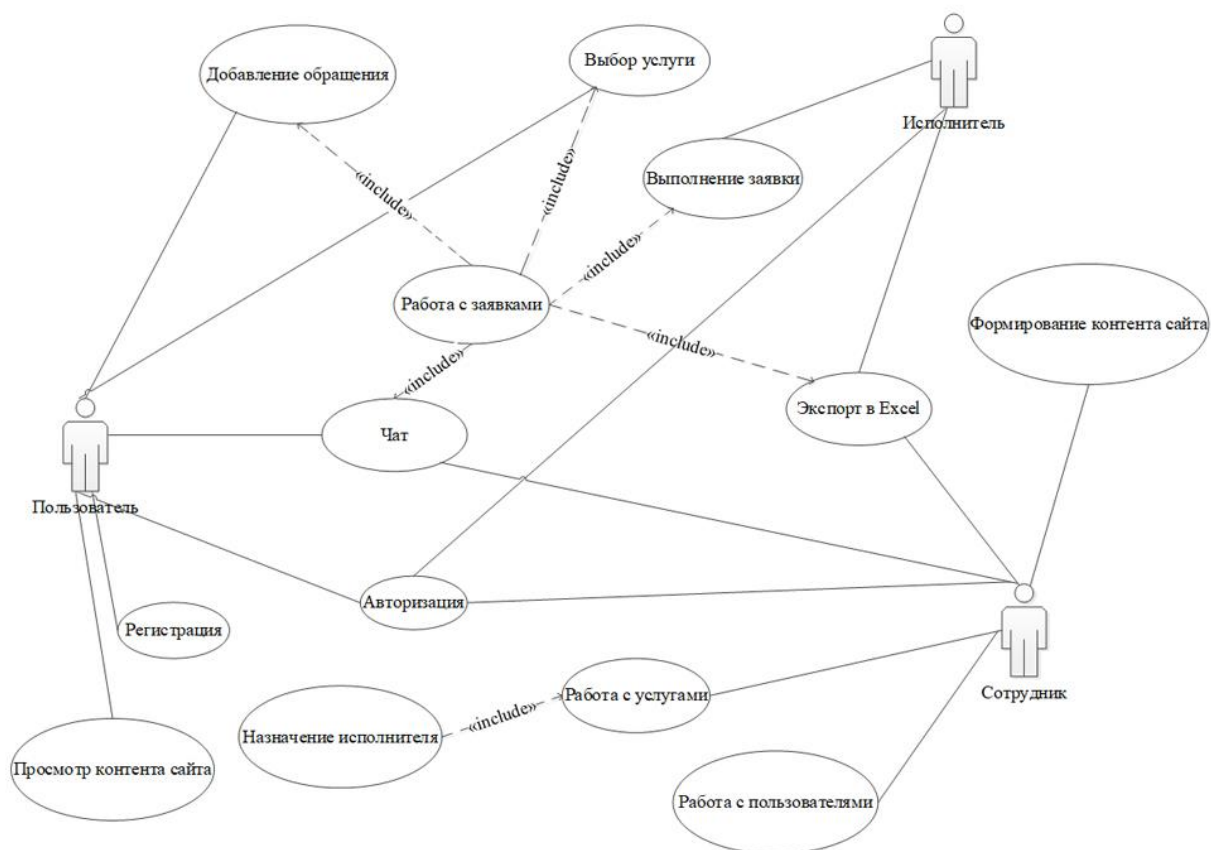


Рисунок 12 – Диаграмма прецедентов

Диаграмма прецедентов содержит следующие варианты использования (прецеденты):

- 1) работа с пользователями (режим сотрудника);
- 2) работа с заявками;
  - выполнение заявки (режим исполнителя);
  - выбор услуги (режим сотрудника);
  - экспорт в Excel (режим ответственного, режим исполнителя);
  - чат (режим сотрудника, режим ответственного, режим клиента);
- 3) регистрация (режим клиента);
- 4) авторизация (режим клиента, режим ответственного, режим исполнителя);
- 5) работа с услугами (режим ответственного);
  - назначение исполнителя (режим ответственного);
- 6) формирование контента (режим ответственного);

7) добавление обращения (режим ответственного);

8) просмотр контента сайта (режим клиента).

Работа программы должна осуществляться в соответствии с Use Case диаграммой, изображённой на рисунке 12 [33].

## 2.2 Проектирование базы данных

В качестве инструментального средства, реализующего метод семантического моделирования данных, было выбрано CASE средство AllFusion ERwin Data Modeler 7.3 [26].

Для того, чтобы вести учет заявок службы техподдержки, выделим информационные объекты – сущности предметной области (таблица 4) [11].

Таблица 4 – Описание сущностей предметной области

Сущности	Описание
Сотрудник	Сотрудники компании, которые обращаются в техподдержку (клиенты)
Исполнитель	Сотрудник отдела ИО, который выполняет заявку
Сущности	Описание
Услуга	Услуги, предоставляемые отделом ИО
Подразделение	Подразделение компании, в котором работает сотрудник
Ответственный	Сотрудник отдела ИО, который принимает заявку и назначает исполнителя
Поддержка	Информация о заявках сотрудников (клиентов)
Чат	Сообщения пользователей и сотрудников
Контент	Это информация, размещенная на страницах сайта

Диаграмма отношений сущностей (ER) – это тип блок-схемы, которая иллюстрирует, как «сущности», такие как люди, объекты или концепции, связаны друг с другом в системе. Диаграммы ER чаще всего используются для проектирования или отладки реляционных баз данных в области разработки программного обеспечения, бизнес-информационных систем, образования и исследований.

Диаграмма «Сущность-Связь» (ER - диаграмма), представлена на рисунке 13.

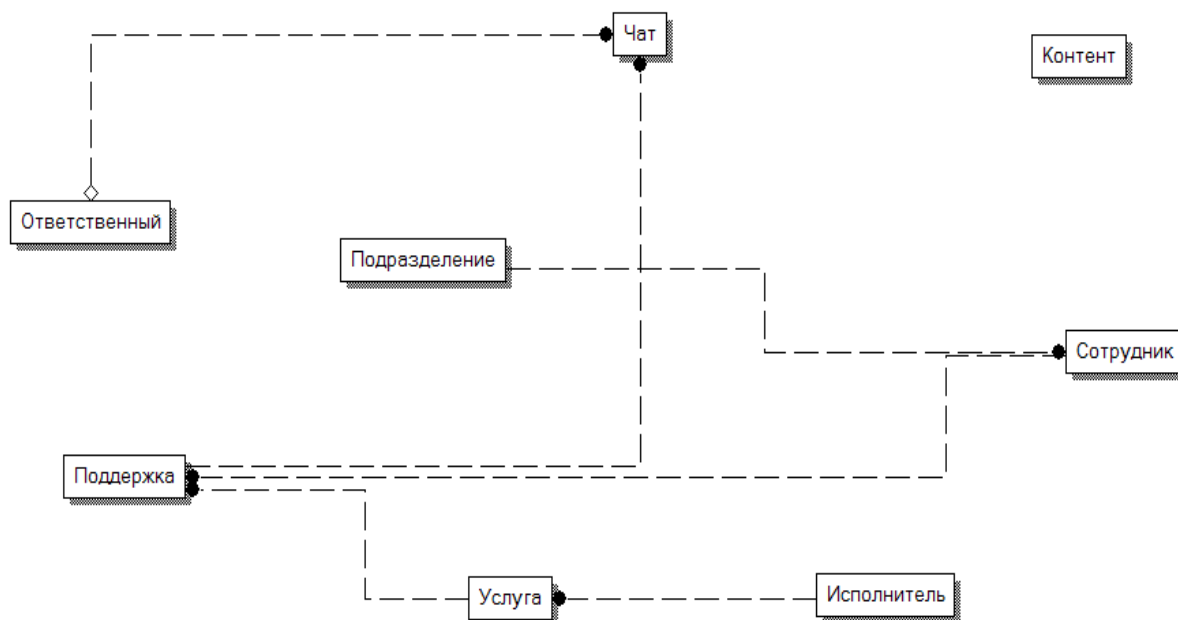


Рисунок 13 – Диаграмма сущностей и связей (ER)

Построение концептуальной модели начинается с определения атрибутов сущностей предметной области и определения ключевых атрибутов – идентификаторов сущностей.

Далее необходимо определить связи между сущностями. Пунктирной линией определяются неидентифицирующие связи – экземпляр дочерней сущности не зависит от родительской сущности, он может присутствовать без экземпляра родительской сущности. В неидентифицирующей связи единственный экземпляр родительской сущности связан с множеством экземпляров дочерней [20].

Атрибуты выделенных сущностей представлены в таблице 5.

Таблица 5 – Описание атрибутов сущностей предметной области

Сущность	Атрибут	Ключи
Сотрудник	Номер сотрудника	Первичный ключ
	Номер подразделения	Внешний ключ (сущность «Подразделение»)
	ФИО	
	Должность	
	Телефон	
	Логин	
	Пароль	
Исполнитель	Номер исполнителя	Первичный ключ
	ФИО	
	Должность	
	Телефон	
	Логин	
	Пароль	
Подразделение Услуга	Номер подразделения	Первичный ключ
	Номер ответственного	Внешний ключ (сущность «Ответственный»)
	Подразделение	
	Описание	
	Номер услуги поддержки	Первичный ключ
	Номер исполнителя	Внешний ключ (сущность «Исполнитель»)
	Услуга	
	Сроки в днях	
	Приоритет («Высокий», «Средний», «Низкий»)	
	Описание	
Чат	Номер сообщения	Первичный ключ
	Номер ответственного (NULL, если сообщение от сотрудника инцидента)	Внешний ключ (сущность «Ответственный»)
	Номер поддержки	Внешний ключ (сущность «Поддержка»)
	Дата и время сообщения	
	Сообщение	
Ответственный	Номер ответственного	Первичный ключ
	ФИО	
	Должность	
	Телефон	
	Логин	
	Пароль	
	Почта	



Продолжение таблицы 5

Сущность	Атрибут	Ключи
Поддержка	Номер заявки	Первичный ключ
	Номер услуги	Внешний ключ (сущность «Услуги»)
	Номер клиента	Внешний ключ, (сущность «Сотрудник»)
	Наименование	
	Дата поддержки	
	Статус (Открыт, Закрыт, Отменен)	
	Описание	
Контент	Номер контента	Первичный ключ
	Наименование	
	Файл	
	Категория (новости, проблемы, документы)	
	Описание	
	Дата размещения	
	Лого (картинка)	

Сущности с атрибутами, первичными и внешними ключами, связями представляют инфологическую модель предметной области, которая графически представлена на рисунке 14.

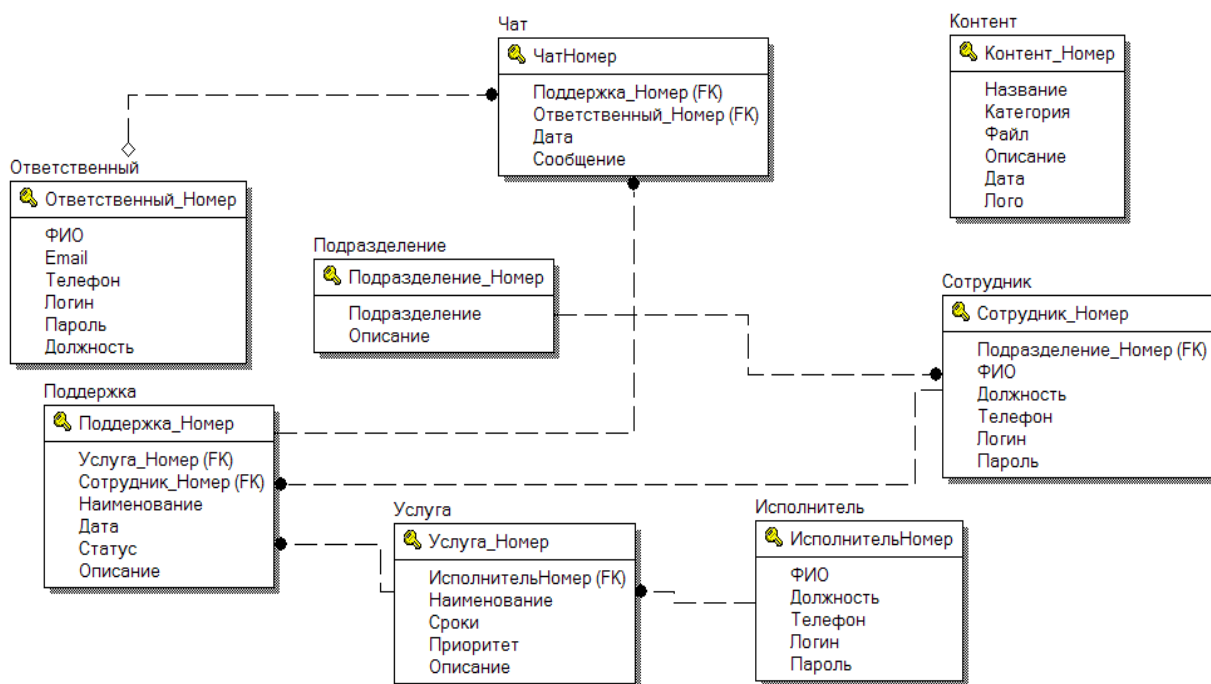


Рисунок 14 – Инфологическая модель данных

Далее определяем типы данных каждого атрибута модели данных согласно выбранной реляционной СУБД, получаем даталогическую модель данных (рисунок 15).

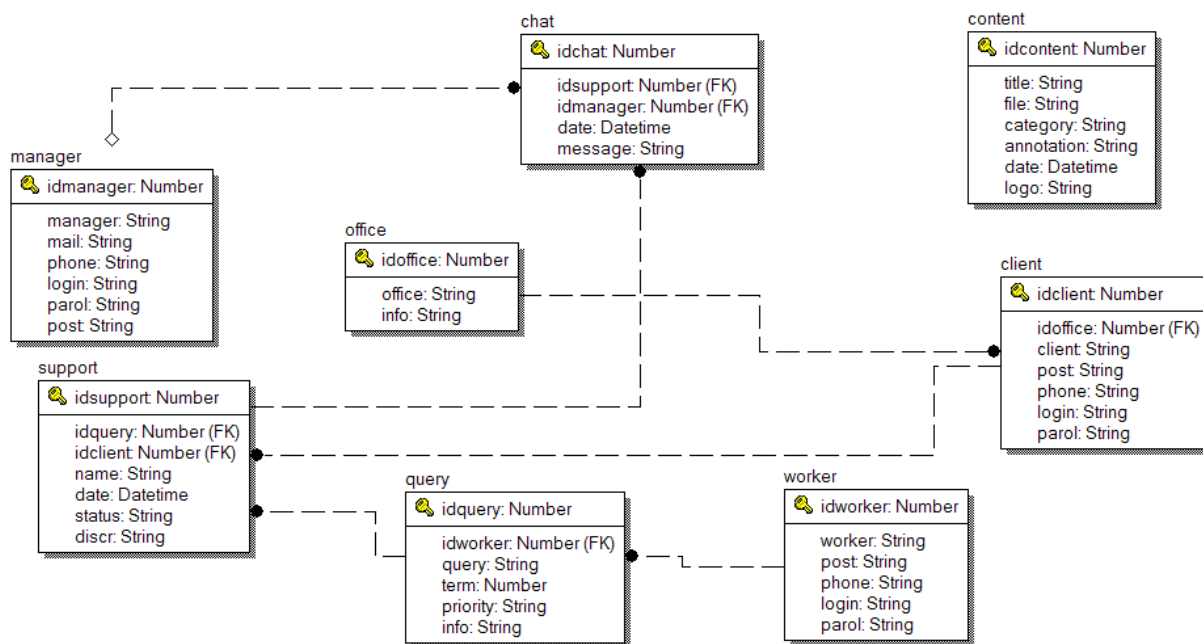


Рисунок 15 – Даталогическая модель данных

Описание данных рассмотрено в таблицах 6-13.

Таблица 6 – Таблица query (запрос в поддержку)

Идентификатор	Описание	Тип данных	Размер	Ключ
idquery	Номер услуги	Int	4	Первичный
idworker	Номер выполняющего	Int	4	Внешний (поле из таблицы worker)
service	Услуга	varchar	80	
term	Сроки в днях	int	4	
priority	Приоритет («Высокий», «Средний», «Низкий»)	varchar	80	
info	Описание	varchar	80	

Таблица 7 – Таблица support (поддержка)

Идентификатор	Описание	Тип данных	Размер	Ключ
idsupport	Номер заявки	Int	4	Первичный
idquery	Номер услуги	Int	4	Внешний (поле из таблицы query)
idclient	Номер клиента	Int	4	Внешний (поле из таблицы client)
name	Наименование	varchar	80	
date	Дата поддержки	date	3	
status	Статус (Открыт, Закрыт, Отменен)	varchar	80	
discr	Описание	varchar	80	

Таблица 8 – Таблица chat (Чат)

Идентификатор	Описание	Тип данных	Размер	Ключ
idchat	Номер организации	int	4	Первичный
idmanager	Номер ответственного (NULL, если сообщение от сотрудника инцидента)	Int	4	Внешний (поле из таблицы manager)
idsupport	Номер поддержки	Int	4	Внешний (поле из таблицы support)
date	Дата и время сообщения	datetime	4	
message	Сообщение	varchar	80	

Таблица 9 – Таблица client (Сотрудник)

Идентификатор	Описание	Тип данных	Размер	Ключ
idclient	Номер сотрудника	Int	4	Первичный
idoffice	Номер подразделения	Int	4	Внешний (поле из таблицы office)
client	ФИО	varchar	80	
post	Должность	varchar	80	
phone	Телефон	varchar	80	
login	Логин	varchar	80	
parol	Пароль	varchar	80	

Таблица 10 – Таблица worker (Исполнитель)

Идентификатор	Описание	Тип данных	Размер	Ключ
idworker	Номер исполнителя	Int	4	Первичный
worker	ФИО	varchar	80	
post	Должность	varchar	80	
phone	Телефон	varchar	80	
login	Логин	varchar	80	
parol	Пароль	varchar	80	

Таблица 11 – Таблица office (Подразделение)

Идентификатор	Описание	Тип данных	Размер	Ключ
idoffice	Номер подразделения	int	4	Первичный ключ
idmanager	Номер ответственного	Int	4	Внешний (поле из таблицы manager)
office	Подразделение	varchar	80	
info	Описание	varchar	80	

Таблица 12 – Таблица content (Контент)

Идентификатор	Описание	Тип данных	Размер	Ключ
idcontent	Номер контента	int	4	Первичный
title	Наименование	varchar	80	
file	Файл	varchar	80	
category	Категория (новости, проблемы, документы)	varchar	80	
annotation	Описание	varchar	80	
date	Дата размещения	date	4	
logo	Лого (картинка)	varchar	80	

Таблица 13 – Таблица manager (Ответственный)

Идентификатор	Описание	Тип данных	Размер	Ключ
idmanager	Номер ответственного	Int	4	Первичный
manager	ФИО	varchar	80	
post	Должность	varchar	80	
phone	Телефон	varchar	80	
login	Логин	varchar	80	
parol	Пароль	varchar	80	
mail	Почта	varchar	80	

Данная модель служит основой для разработки базы данных в выбранной СУБД.

### 2.3 Информационное и техническое обеспечение web-приложения

Информационная модель показывает хранилище данных, потоки информации между блоками данных и результирующую информацию. Информационная модель представлена на рисунке 16 [9].

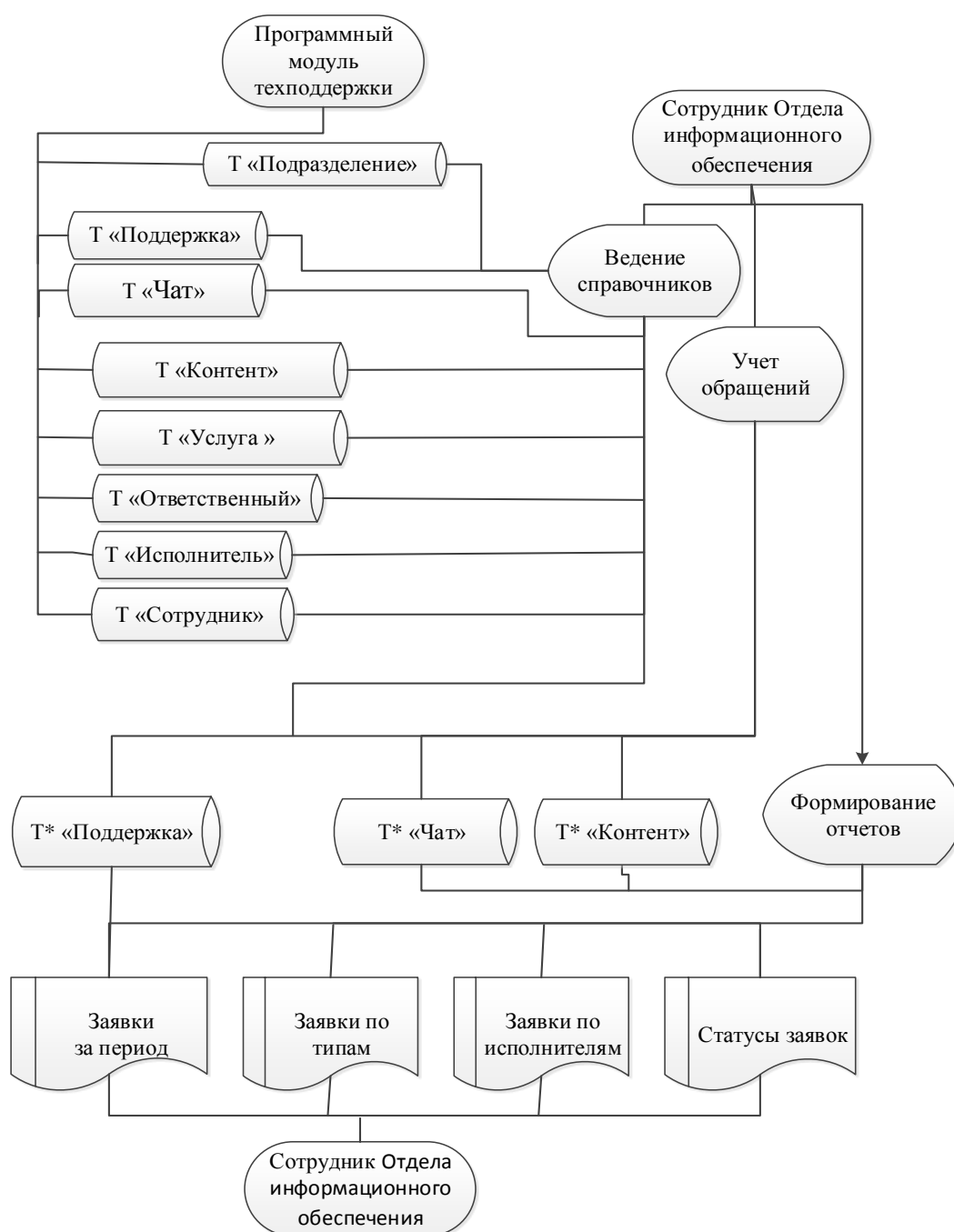


Рисунок 16 – Информационная модель системы

Основная последовательность работы в системе следующий.

Сотрудник Отдела информационного обеспечения первоначально заполняет справочники «Сотрудник», «Исполнитель», «Услуга», «Подразделение», «Ответственный» на основании регламента услуг и штатного расписания.

Далее формируются справочники «Поддержка», «Чат», «Контент».

В ходе формирования справочника «Сотрудник» каждый пользователь получает логин и пароль для доступа в систему. Пользуясь эти данными, пользователь получает доступ с к странице подачи заявки, где, используя справочники и информацию по услугам от сотрудников, изменяет содержание таблиц «Поддержка», «Чат». Также ответственный сотрудник следит за обновлением контента сайта, заполняя таблицу «Контент».

Эти таблицы отображаются у сотрудника Отдела информационного обеспечения, который, на основе данных о выполнении заявки, вносит дату ее выполнения и статус.

Локально-вычислительная сеть ГАУИО «Ирэкспертиза» включает пользовательский и серверный сегменты. Серверный сегмент имеет два коммутационных шкафа, один из которых размещен в серверной комнате, а второй – в отделе информационного обеспечения.

Техническое и программное обеспечение ГАУИО «Ирэкспертиза» и требования к его обновлению представлено в таблице 14.

Таблица 14 – Анализ аппаратного и программного обеспечения в организации

№	Техническое/программное обеспечение	Требует обновления (Да/Нет)
Техническое обеспечение		
1	Коммутационный шкаф серверной комнаты: Сервера IBM BladeCenter PS703 Express (сервер приложений, файл-сервер, сервер БД). Маршрутизатор Mikrotik RB1100AH. Дисковый массив HP StorageWorks P2000 G3 MSA, до 58 ТБ. В коммутационном шкафу в отделе информационного обеспечения размещен файловый сервер IBM System x3650.	Нет

Продолжение таблицы 14

№	Техническое/программное обеспечение	Требуется обновления (Да/Нет)
2	Рабочие станции сотрудников ГАУИО «Ирэкспертиза»: Dell Precision 3430: процессор: Intel Core i5; память: 8 Гбайт (1 x 8 Гбайт) памяти DDR4; жесткий диск: 1 Тбайт, 7 200 об/мин, 2,5 дюйма; графика NVIDIA Quadro P1000; сетевая карта: 10/100 Мбит/с	Да, на Рабочую станцию Lenovo - ThinkStation P330, Intel Core i7 8700 3200MHz, память DIMM DDR4 8GB, HDD 1TB, графика NVIDIA Quadro P1000; сетевая карта: 10/100 Мбит/с, DVD-RW
3	Рабочие станции в отделе информационного обеспечения: ASUS E500 G5: Чипсет: Intel C246 Chipset; Процессор: Intel Xeon processor E-2100; Память: 32 Гбайт (4 x 8 Гбайт) памяти DDR4; Жесткий диск: 2 Тбайт, 7 200 об/мин, 2,5``; Графика NVIDIA Quadro P1000; Сетевая карта: 10/100 Мбит/с.	Нет
4	В кабинетах установлены маршрутизаторы 3COM Router3000	Нет
5	В отделе информационного обеспечения установлены маршрутизаторы D-Link DIR-620/GA.	Нет
6	Клавиатура+мышь проводная Ritmix RKC-010 черный, соединение USB	Нет
7	В пользовательском сегменте имеются сетевые лазерные ЧБ МФУ Canon IR1024iF и ИБП APC	Нет
8	В отделе информационного обеспечения имеется лазерный цветной МФУ Canon imageRUNNER C1028iF.	
9	Сканер Canon CanoScan LiDE 220, планшетный сканер, CIS, A4, 4800x4800 dpi, USB 2.0	Нет
10	Для передачи данных по сети используется кабель витая пара 5 категории (100BASE-TX) и категории 5е (1000BASE-T) со скоростью передачи до 100 мбит/сек для 100BASE-TX и 1000 мбит/сек для 1000BASE-T.	Нет
<b>Программное обеспечение</b>		
1	Microsoft Windows Server 2019 – серверная операционная система от Microsoft. Система является частью семейства Windows NT.	Нет
2	Microsoft Windows 10 – операционная система от Microsoft, используется для поддержки работы компьютерной системы: связи между программной и аппаратной частями компьютера.	Нет

## Окончание таблицы 14

№	Техническое/программное обеспечение	Требует обновления (Да/Нет)
3	Subversion – это система контроля версий с открытым исходным кодом существует для того, чтобы быть общепризнанной и принятой в качестве централизованной системы управления версиями с открытым исходным кодом, характеризующейся своей надежностью в качестве безопасного убежища для ценных данных; простота модели и использования; и его способность поддерживать потребности широкого круга пользователей и проектов, от отдельных лиц до крупномасштабных корпоративных операций.	Нет
4	«Symantec Endpoint Security» клиентская часть, позволяющая обеспечить защиту ПК от вирусов	Нет
5	«Microsoft Office 2016» – программное обеспечение для создания и ведения электронных документов	Нет
7	Opera, Google Chrome – быстрые и удобные браузеры для отображения веб-страниц	Нет
9	Антивирус Касперский 6 – мощный антивирусный пакет программ	Нет
10	«1С:Бухгалтерия 8 (редакция 3.0)» на базе платформы «1С:Предприятие 8.3».	Да, периодически, при обновлении программного продукта на сайте компании 1С
11	В качестве сервера баз данных используется MS SQL Server 2019 Standart Edition.	Нет
12	Adobe Acrobat 6.0 Professional – программа для создания и чтения файлов в pdf-формате	Нет
13	Справочная система Консультант Плюс – для поддержки отдела правового сопровождения	Да, периодически, при обновлении программного продукта на сайте компании Консультант Плюс

### 2.4 Построение архитектуры программного модуля техподдержки

Как только пользователь открывает веб-страницу, сервер отправляет браузеру определенные данные в ответ на запрос пользователя. Чтобы быть точным, веб-клиент (или пользовательский агент) может запрашивать веб-ресурсы или более распространенные веб-документы (HTML, JSON, PDF и т.



д.) через веб-сервер. Затем при этих минимальных манипуляциях появляется запрошенная информация. После этого начинается взаимодействие между пользователем и сайтом.

Веб-браузеры взаимодействуют с веб-серверами, используя протокол передачи гипертекста (HTTP). Когда нажимаете ссылку на веб-странице, отправляете форму или выполняете поиск, HTTP-запрос отправляется из браузера на целевой сервер [13].

Запрос включает URL-адрес, идентифицирующий затронутый ресурс, метод, который определяет требуемое действие и может включать дополнительную информацию, закодированную в параметрах URL-адреса, данные, отправленные методом HTTP POST, или связанные файлы cookie.

Веб-серверы ждут сообщений о запросах клиентов, обрабатывают их, когда они поступают, и отвечают на веб-браузер с сообщением об ответе HTTP. Ответ содержит строку состояния, указывающую, удалось ли выполнить запрос.

Упрощенная схема таких работ показана на рисунке 17.

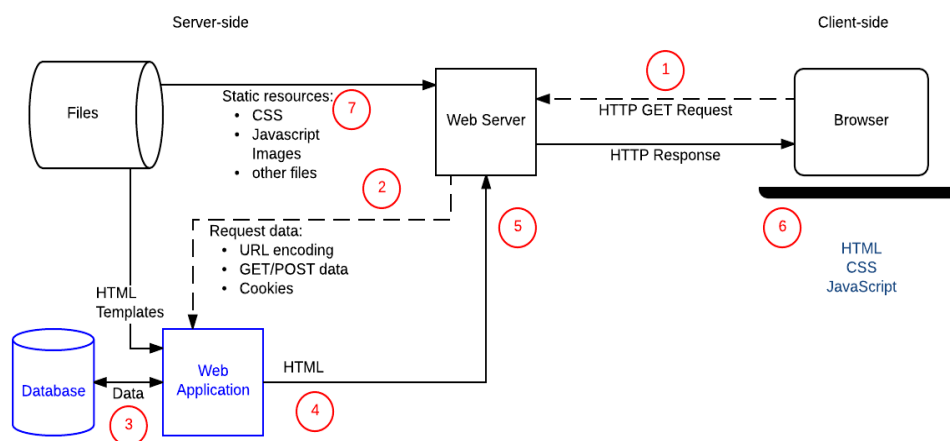


Рисунок 17 – Схема работы динамического сайта

Запросы (2) отправляются на сервер. Сервер обрабатывает запрос, считывает требуемую информацию из базы данных (3), обрабатывает данные и HTML-шаблоны (4) и отправляет ответ, содержащий сгенерированную страницу (5,6).

Передача информации в компании осуществляется при помощи локальной сети, и файлового хранилища. Сеть построена на основе сервера («клиент – сервер») – один из самых организованных и востребованных вариантов построения сети. Характеризуется высокой скоростью передачи данных, высоким уровнем безопасности и организации сети.

Система управления базами данных и необходимое ПО установлено на сервере и не требует вмешательства.

Обработка и хранение информации будет происходить с помощью сайта, для удаленной работы необходим доступ в Интернет, поэтому все компьютеры имеют прямой доступ к сети без ограничений в соответствии со всеми требованиями информационной безопасности.

Функциональная структура web-приложения с учётом клиент-серверной архитектуры представлена на рисунке 18 [14].

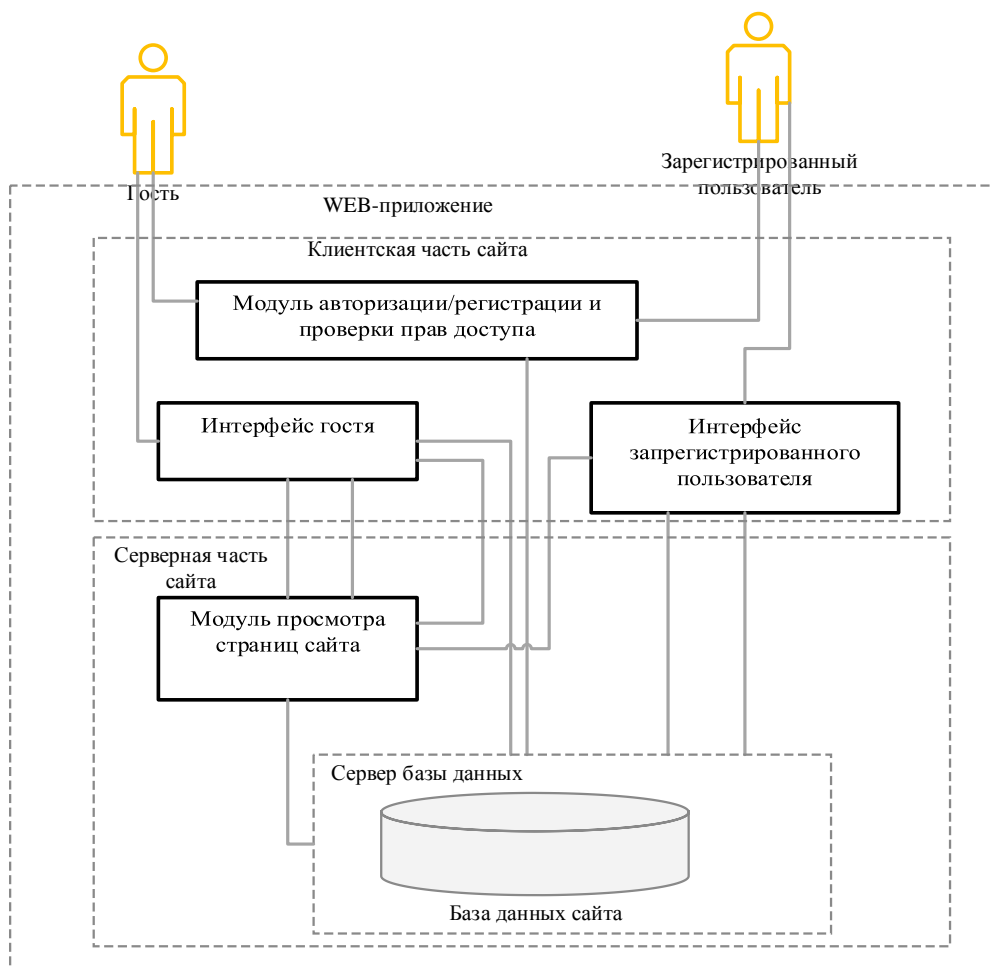


Рисунок 18 – Функциональная структура web-приложения

После анализа аналогичных по возможностям информационных систем было определено, что для реализации ИС прежде всего потребуются организация веб-сервера. При разработке любых веб-приложений или сайтов обычно используется локальный сервер, представляющий собой комплекс программ, устанавливаемый на любой компьютер и используемый либо локально, либо в пределах локальной сети. Для реализации полноценного, работающего в сети интернет сервера можно использовать либо хостинг у провайдера, либо реализовать собственный выделенный сервер.

Как локальный, так и полноценный сервер обычно состоит из базового набора программ. При разработке модуля техподдержки выбран веб-сервер Apache, отличающийся своей гибкостью в конфигурирование и расширяемости. Расширение происходит за счет подключаемых модулей. В качестве серверного языка программирования в основном используются: Java, PHP, Perl, APS.NET, Ruby. Наиболее популярными серверами базы данных для подобных задач являются MySQL и PostgreSQL [20].

В качестве модели доступа к данным выбрана трехуровневая клиент-серверная модель. Такая архитектура обеспечивает большую масштабируемость, большую конфигурируемость и более широкие возможности по обеспечению безопасности и отказоустойчивости (рисунок 19) [15].

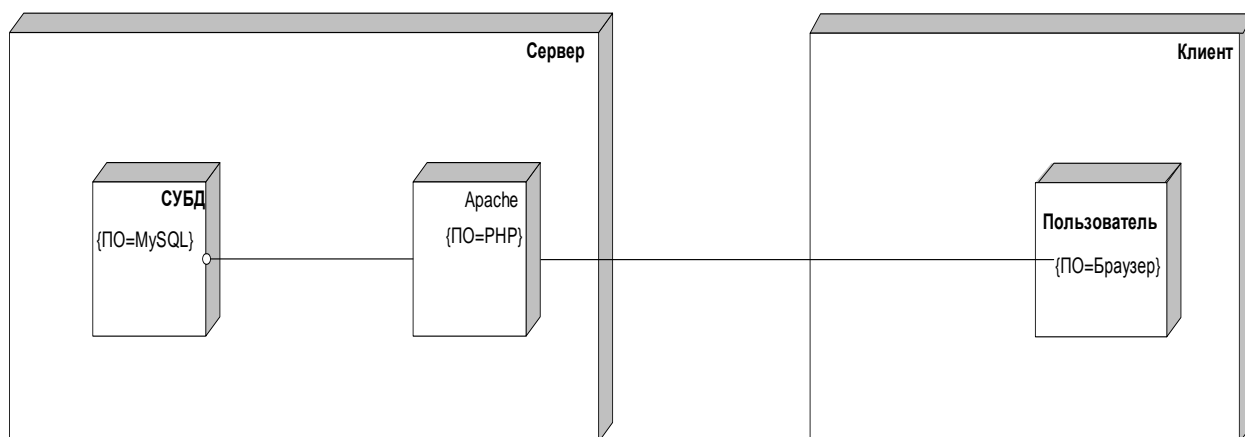


Рисунок 19 Диаграмма размещения

## 2.5 Разработка серверной части модуля техподдержки

Установим программный пакет Денвер (рисунок 20) [27].

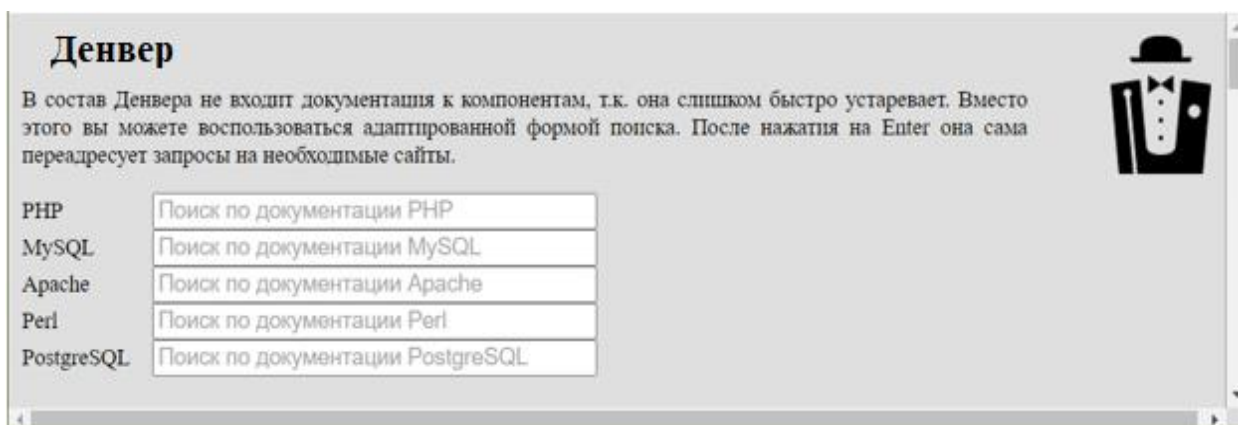


Рисунок 20 – Запуск Денвер

Для работы с базой данных используем phpmyadmin.

Создаем в phpmyadmin новую базу данных support (рисунок 21) [29].

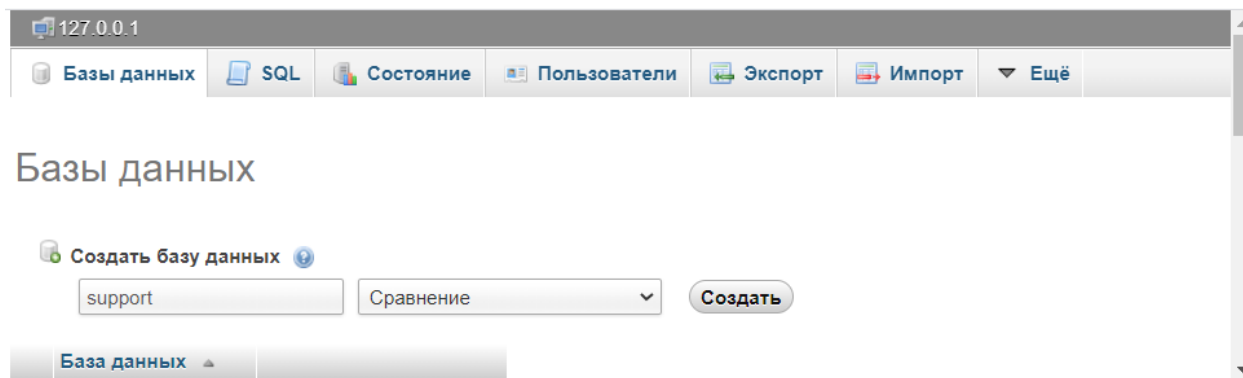


Рисунок 21 – Новая база данных

Импортируем таблицы и данные в созданную пустую базу данных из созданного файла скрипта support.sql (рисунок 22).

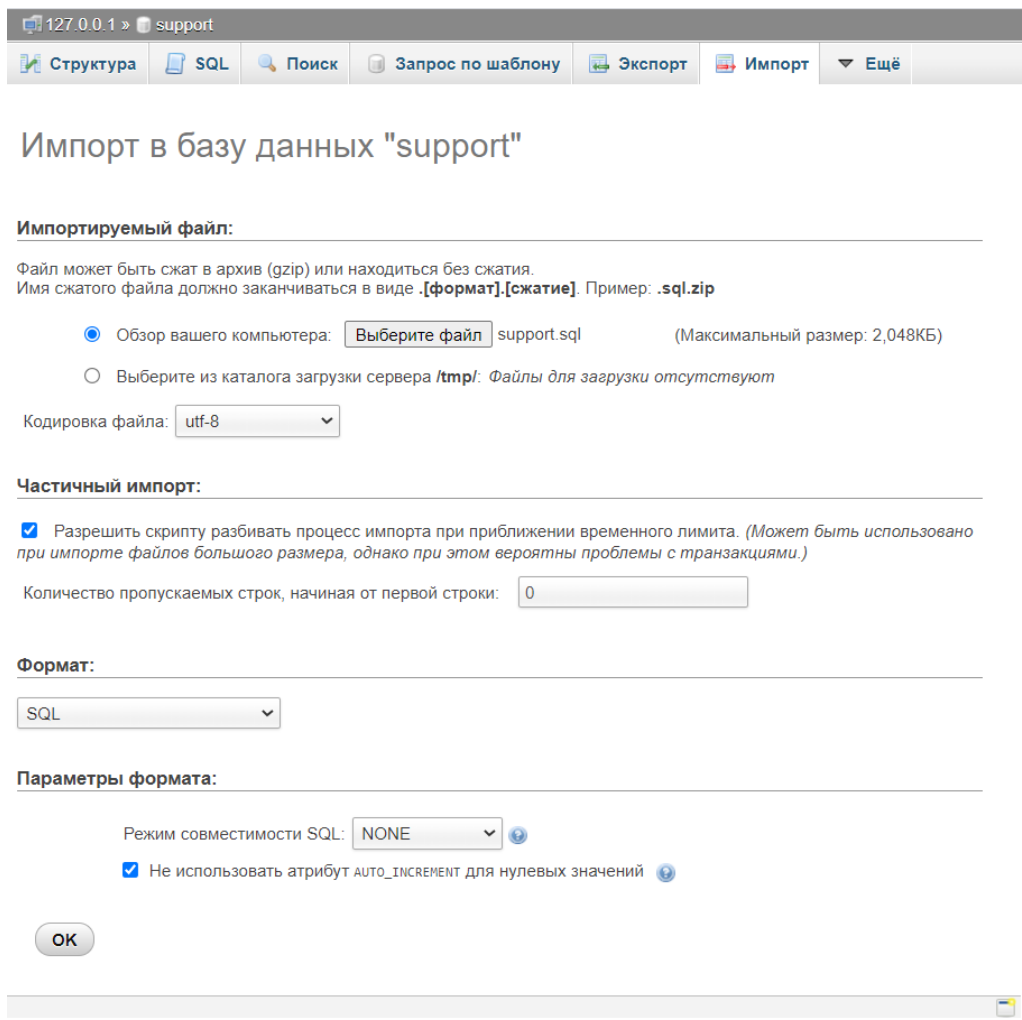


Рисунок 22 – Подготовка импорта из файла скрипта

Структуру базы данных представим на рисунке 23.

Таблица	Действие	Строки	Тип	Сравнение	Размер	Фрагментировано
<input type="checkbox"/> chat	Обзор Структура Поиск Вставить Очистить Удалить	4	InnoDB	cp1251_bin	48 КБ	
<input type="checkbox"/> client	Обзор Структура Поиск Вставить Очистить Удалить	1	InnoDB	cp1251_bin	32 КБ	
<input type="checkbox"/> content	Обзор Структура Поиск Вставить Очистить Удалить	6	InnoDB	cp1251_bin	16 КБ	
<input type="checkbox"/> manager	Обзор Структура Поиск Вставить Очистить Удалить	1	InnoDB	cp1251_bin	16 КБ	
<input type="checkbox"/> office	Обзор Структура Поиск Вставить Очистить Удалить	1	InnoDB	cp1251_bin	16 КБ	
<input type="checkbox"/> query	Обзор Структура Поиск Вставить Очистить Удалить	5	InnoDB	cp1251_bin	32 КБ	
<input type="checkbox"/> support	Обзор Структура Поиск Вставить Очистить Удалить	7	InnoDB	cp1251_bin	48 КБ	
<input type="checkbox"/> worker	Обзор Структура Поиск Вставить Очистить Удалить	2	InnoDB	cp1251_bin	16 КБ	
<b>8 таблиц</b>	<b>Всего</b>	<b>27</b>	<b>InnoDB</b>	<b>utf8_general_ci</b>	<b>224 КБ</b>	<b>0 Байт</b>

Рисунок 23 – Структура базы данных support

В дизайнерае MySQL формируется диаграмма базы данных (рисунок 24).

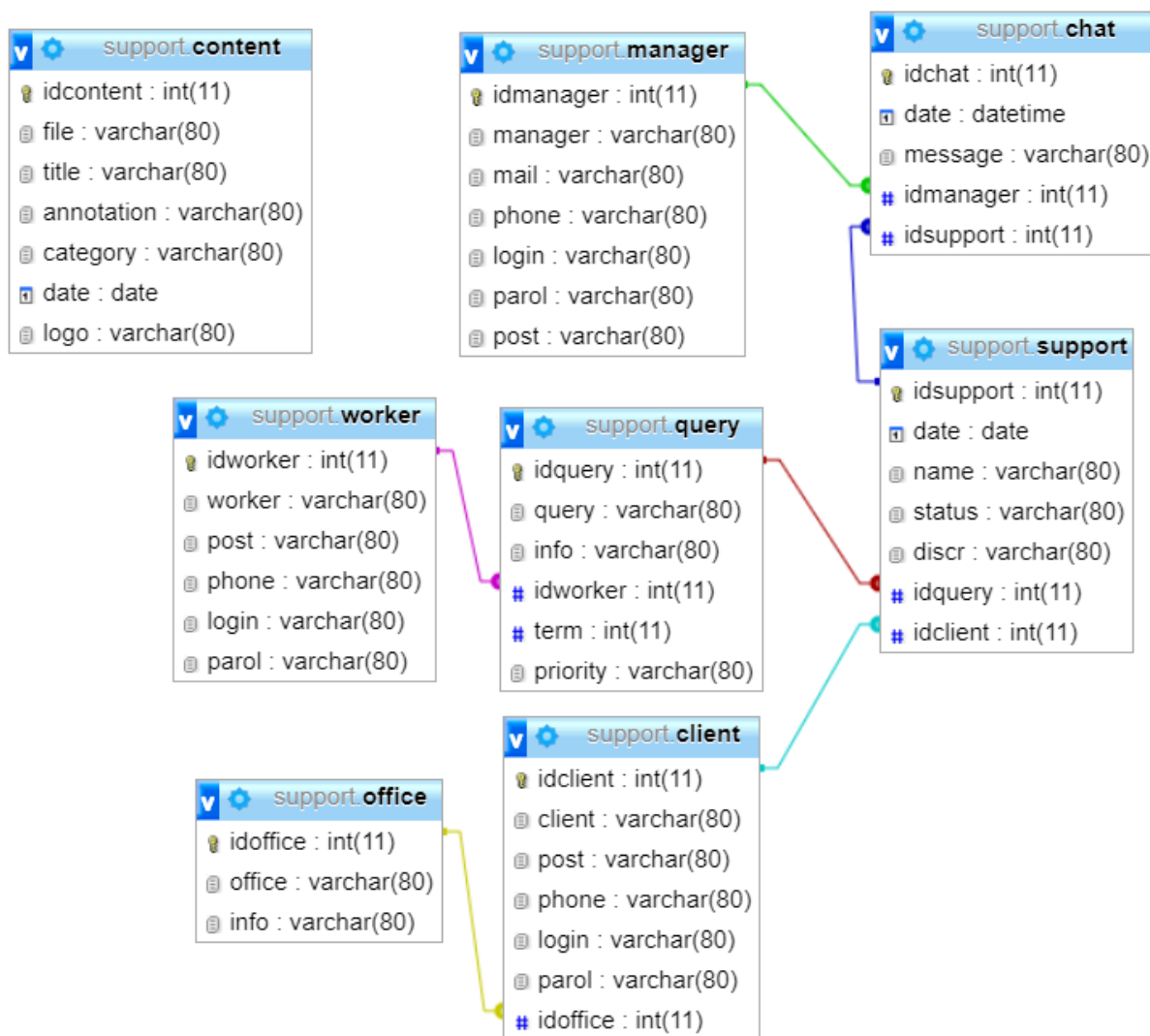


Рисунок 24 – Диаграмма базы данных support в дизайнерае MySQL

## 2.6 Разработка клиентской части модуля техподдержки

Рассмотрим требования к программному обеспечению клиентской части модуля техподдержки:

- компьютер с процессором рекомендуется от 1.5ГГц;
- оперативная память 256 Мб (рекомендуется от 512 Мб).

Страница должна максимально идентично отображаться во всех современных браузерах: Microsoft Edge, Mozilla FireFox, Opera и Google Chrome последних версий и при разрешениях монитора от 1024x768 до 1920x1080.

Необходимо выбрать адрес сайта. Именно по данному адресу посетители будут видеть данный сайт в Интернете. Придумываем свободный

адрес сайта: «support.ru» [16].

Сайт будет состоять из следующих основных файлов (рисунок 25).

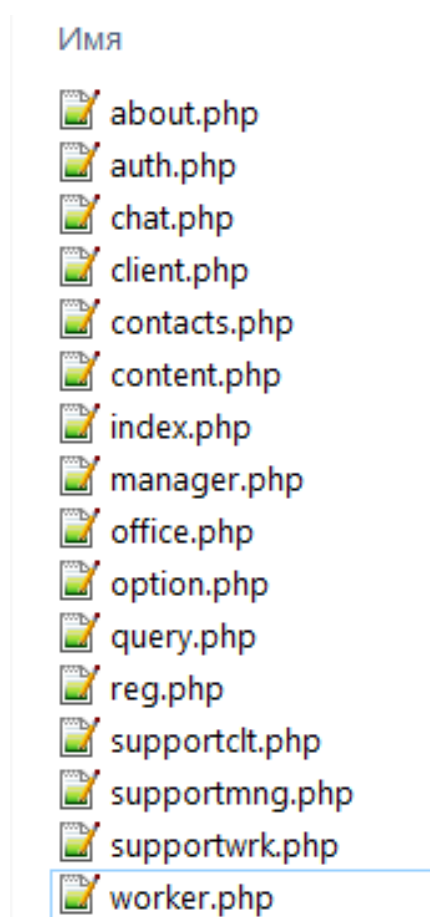


Рисунок 25 – Структура папок сайта

В рамках данного проекта была вручную вставлена необходимая информация в соответствующие теги страниц в текстовом редакторе Notepad++ [30].

Были созданы следующие файлы для компоновки страниц сайта:

- about.html – web-страница с информацией о службе техподдержки;
- auth.php – web-страница для авторизации пользователей в системе;
- chat.php – web-страница для обеспечения чата в рамках поддержки;
- client.php – web-страница для работы со справочником сотрудников;
- contacts.html – web-страница с контактной информацией службы техподдержки;
- content.php – web-страница для работы с контентом сайта;

- index.php - основная страница web-сайта, которая содержит главное меню;
- manager.php – web-страница для работы со справочником ответственных лиц;
- office.php – web-страница для работы со справочником подразделений;
- option.php – web-страница, которая содержит процедуры и функции подключения к базе данных;
- query.php – web-страница для работы с услугами техподдержки;
- reg.php – web-страница для регистрации пользователей в системе;
- supportclt.php – web-страница для работы пользователей с поддержкой;
- supportmng.php – web-страница для работы ответственных лиц с поддержкой;
- supportwrk.php – web-страница для работы исполнителей с поддержкой;
- worker.php – web-страница для работы со справочником исполнителей.

В качестве компонентов (модулей) выступают файлы программных модулей и файл базы данных.

В разработанной системе предусмотрено два вида пользователей;

- сотрудник Отдела информационного обеспечения;
- пользователь.

Диаграмма компонентов программного модуля техподдержки представлена на рисунке 26.



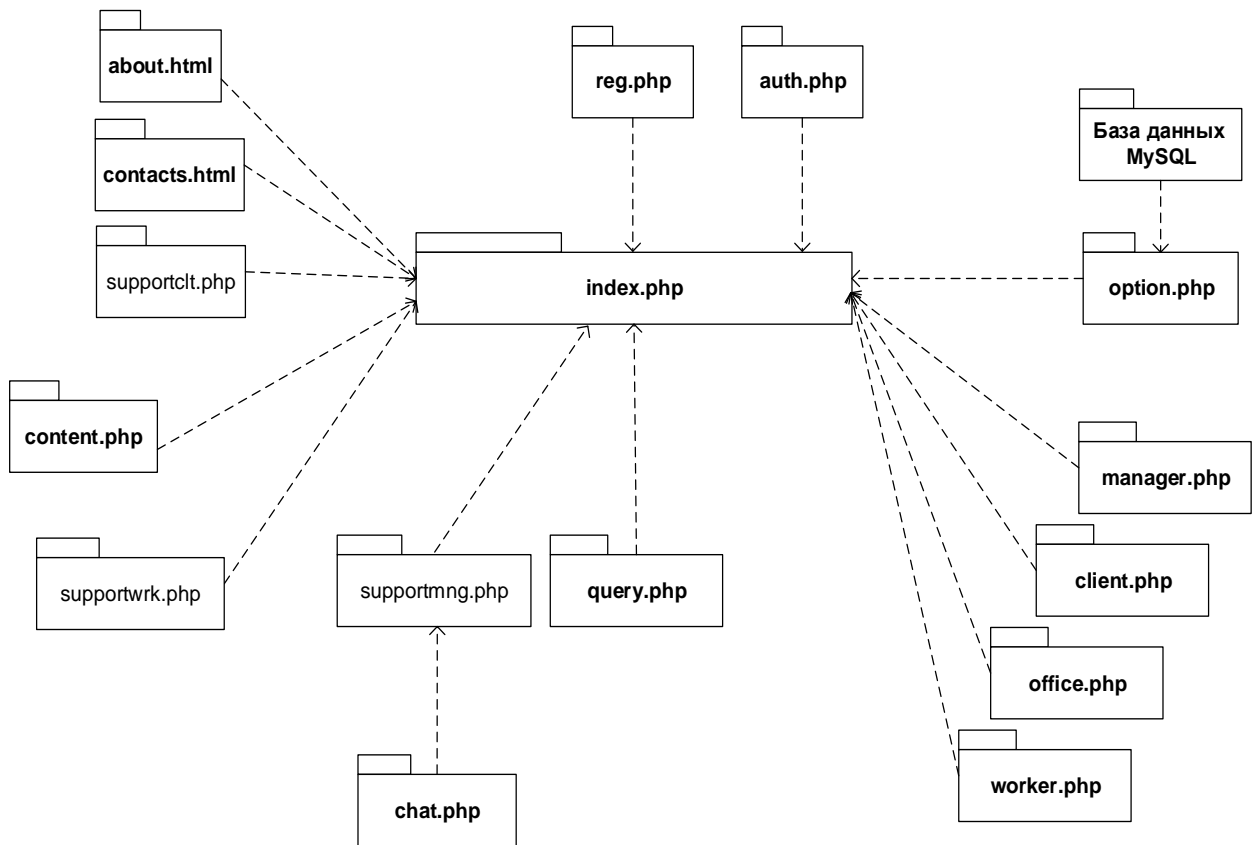


Рисунок 26 – Диаграмма компонентов

Дерево функций системы представлено на рисунке 27 [24].

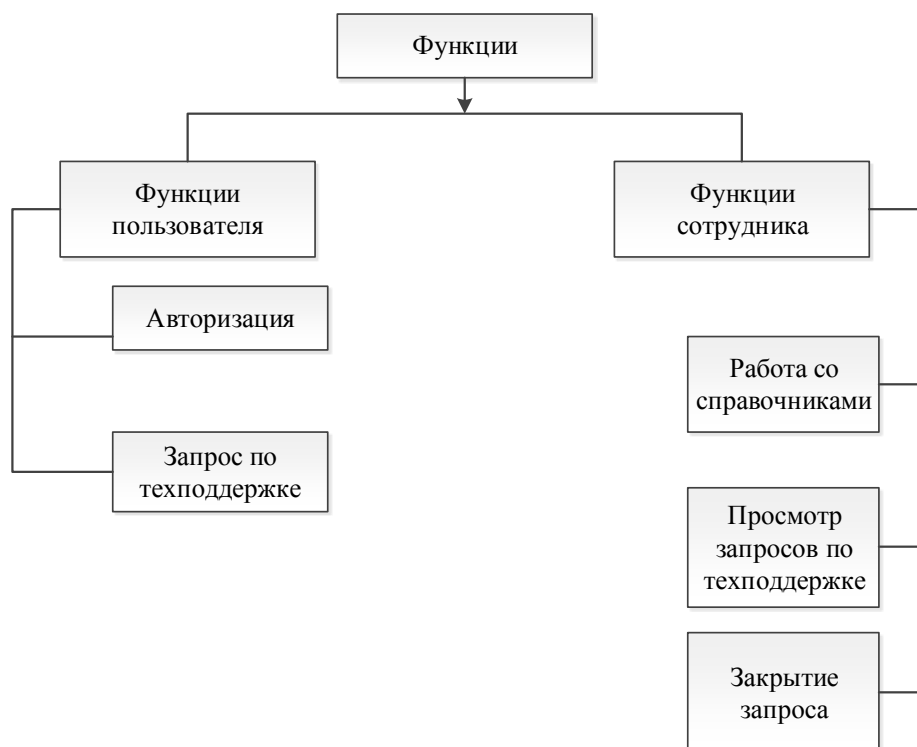


Рисунок 27 – Дерево функций системы для пользователя и сотрудника

Здесь под работой со справочниками подразумевается их заполнение и редактирование, под закрытием запроса – внесение данных о выполненных работах по запросу пользователя.

Сценарии диалога для пользователя представлен на рисунке 28.



Рисунок 28 – Сценарий диалога для пользователя

Сценарий диалога для сотрудника Отдела информационного обеспечения представлен на рисунке 29.



Рисунок 29 – Сценарий диалога для сотрудника Отдела информационного обеспечения

Экранные формы интерфейса представлены в приложении А.

Выводы по 2 главе

В данной главе выполнено проектирование системы, рассмотрено информационное обеспечение задачи, выполнено моделирование базы данных.

Далее выполнено описание процесса проектирования информационного программного обеспечения задачи, в приложении А представлены экранные формы, демонстрирующие работу системы и их описание.

## Глава 3 Обоснование экономической эффективности разработки и внедрения web-приложения

### 3.1 Выбор и обоснование методики расчёта экономической эффективности разработки и внедрения web-приложения

Себестоимость web-приложения определяем в виде суммы затрат на разработку, внедрение, техническое обслуживание и эксплуатацию оборудования, которое будет использовано при разработке и внедрении программного обеспечения.

Стоимость разработки и внедрения рассчитаем по формуле в рублях (1).

$$C_p = Z_p + Z_{\text{раз}} + Z_{\text{вн}} + P_o + H_p, \quad (1)$$

где  $Z_p$  – заработная плата разработчика;

$Z_{\text{раз}}$  – затраты на разработку;

$Z_{\text{вн}}$  – затраты на внедрение;

$P_o$  – затраты на оборудование, руб.;

$H_p$  – накладные расходы (10% от  $Z_p$ ).

Сумма расходов на оборудование в рублях:

$$P_o = C_{\text{эн}} + A_o + C_{\text{рем}}, \quad (2)$$

где  $C_{\text{эн}}$  – затраты на электроэнергию;

$A_o$  – амортизация;

$C_{\text{рем}}$  – затраты на ремонт оборудования;

Затраты на электроэнергию:

$$C_{\text{эн}} = M_{\text{пр}} \cdot T_m \cdot C_{\text{квт/ч}}, \quad (3)$$

где  $M_{\text{пр}}$  – количество используемой электроэнергии, кВт/час;

$T_m$  – машинное время (50% от  $T_{\text{пр}}$ );

$C_{\text{квт/ч}}$  – тариф, (2,28 руб.)

Амортизация:

$$A_o = \frac{H_a \cdot C_{об} \cdot T_m}{100\% \cdot \Phi_d}, \quad (4)$$

где  $H_a$  – годовая норма амортизации, %:

$$H_a = \frac{1}{T_n} \cdot 100\%. \quad (5)$$

$C_{об}$  – стоимость оборудования, руб.;

$T_n$  – срок службы по нормативам, лет;

$\Phi_d$  – фонд рабочего времени оборудования в год, час.

$$\Phi_d = ((365 - C - B - Пр) \cdot 8 - Пр \cdot 1) \cdot S \cdot (1 - a/100), \quad (6)$$

где 365 – количество дней в году;

$C, B, Пр$  – нерабочие дни в году: суббота, воскресенье и праздники;

8 – смена, ч;

$S$  – количество смен в сутки;

$a$  – потери времени на ремонт оборудования ( $a = 3-5\%$ ).

Затраты на ремонт оборудования:

$$C_{рем} = \frac{H_p \cdot C_{об} \cdot T_m}{100\% \cdot \Phi_d}, \quad (7)$$

где  $H_p$  – средства на ремонт оборудования (принимается 2-4% от стоимости);

$C_{об}$  – стоимость оборудования, руб.

Экономия трудоемкости работ в часах:

$$\Delta T = T_б - T_n, \quad (8)$$

где  $T_б$  – трудоемкость по базовому варианту;

$T_n$  – трудоемкость по новому варианту.

Капитальные затраты на разработку:

$$K = \frac{T_{мв} * K_k}{\Phi_d} + C_p, \quad (9)$$

где  $K_k$  – капитальные затраты на компьютер, руб.;

$\Phi_d$  – фонд полезного рабочего времени оборудования в год без ремонта, час/год;

$T_{\text{МВ}}$  – машинное время для нового варианта работы, машино-час/год;

$C_p$  – себестоимость используемой программы, руб.

Затраты на внедрение  $Z_v$ :

$$Z_v = Z_y + Z_t \quad (10)$$

где  $Z_y$  – затраты на установку;

$Z_t$  – затраты на тестирование

Общие затраты в год на разработку и эксплуатацию web-приложения в рублях:

$$C = K + C_1 \quad (11)$$

где  $K$  - капитальные затраты на разработку;

$C_1$  – стоимостные затраты по новому варианту

### 3.2 Расчёт показателей экономической эффективности

Выполним расчеты, используя длительность работ на этапе проектирования (таблица 15).

Таблица 15 – Примерная длительность работ на этапе проектирования

Наименование работ	Длительность работ (дней)		
	Минимум (Топт)	Максимум (Тпес)	Ожидаемая
Разработка технического задания	9	10	9,5
Сбор данных	6,5	8	7
Набор кодов web-приложения*	9	10	9,5
Отладка кодов web-приложения*	2,5	4	3
Проведение тестирования кодов web-приложения *	0,5	1	1
Итого	27,5	33	30
Примечание: * работы, производимые с помощью компьютера.			

Рабочий день: 8 часов.

Всего на разработку web-приложения: 30 дней=240 часов.

На разработку web-приложения с использованием компьютера:

13,5 дней=108 часов.

Таблица 16 – Данные для расчета себестоимости web-приложения

Показатель	Ед.изм.	Значения
Трудоемкость создания программы	час	240
Трудоемкость работ на ПЭВМ	час	108
Месячная заработная плата техника-программиста	руб.	25000
Потребляемая энергия	квт/ч	0,4
Стоимость электрической энергии, за 1 кВт	руб.	5
Годовая норма амортизации	%	20
Стоимость оборудования	руб.	55000
Фонд рабочего времени оборудования	час	1860

Выполним расчет расходов на содержание и эксплуатацию оборудования при разработке web-приложения. Расчеты выполним по формулам из методики расчета с помощью программы Excel и сведем в таблицу (таблица 17).

Таблица 17 – Данные для расчета себестоимости web-приложения

Виды затрат	Сумма (руб.)
Затраты на электроэнергию, $C_{ЭН}$	216
Амортизация, $A_0$	639
Затраты на ремонт компьютера, $C_{рем}$	96
Расходы оборудование при разработке web-приложения	951
Расходы на оборудование при разработке программы с учетом прочих расходов (2% от суммы)	1142

Затраты на разработку сайта сведем в таблицу (таблица 18).

Таблица 18 – Расчет затрат на разработку

Виды работ	Сумма (руб.)
Разработка дизайна web-приложения: макетов дизайна главной и внутренних страниц сайта.	500
Реализация меню web-приложения.	500
Подготовка таблиц-шаблонов web-приложения.	500
Наполнение страниц web-приложения контентом: текст и рисунки	500
Программная разработка web-приложения: программирование динамических элементов сайта	2000
Итого	4000

Затраты на установку и тестирование сайта сведем в таблицу (таблица 19).

Таблица 19 – Затраты на установку и тестирование

Виды затрат	Сумма (руб.)
Затраты на установку, $Z_y$	500
Затраты на тестирование, $Z_t$	500
Итого, Затраты на внедрение $Z_b$	1000

Себестоимость web-приложения равна:

$$C_p = 25000 + 4000 + 1000 + 1142 + 2500 = 33642 \text{ руб.}$$

Далее выполним расчет необходимых капитальных вложений.

Капитальные вложения составят:

$$K = \frac{108 \cdot 55000}{1860} + 33642 = 36835,55 \text{ руб.}$$

Стоимость эксплуатации ( $C_e$ ) представлена в таблице 20.

Таблица 20 – Расчет затрат на эксплуатацию системы

Виды работ	Количество затрат времени в месяц	Стоимость машинного времени, руб./час	Всего, руб.
Работы по поддержанию системы в рабочем состоянии			
Продолжение таблицы 20	1 рабочих дня (8 часа)	100	3200
Удаление ненужных данных	1 рабочий день (8 часов)	100	800
Добавление новых данных	1 рабочий день (8 часов)	100	800
Обеспечение ежемесячной защиты	2 рабочих дня (16 часов)	100	1600
Резервное копирование	1 рабочий день (8 часов)	100	800
Итого в месяц	72 часа		7200
Итого в год	864 часа		86400

Расчет расходов на содержание и эксплуатацию оборудования при использовании web-приложения выполним с помощью программы Excel и сведем в таблицу (таблица 21).



Таблица 21 – Расходы на содержание и эксплуатацию оборудования при использовании web-приложения

Виды затрат	Сумма (руб.)
Затраты на электроэнергию, $C_{ЭН}$	350,6
Амортизация, $A_0$	5110
Затраты на ремонт компьютера, $C_{рем}$	766,5
Расходы оборудование при работе web-приложения	6227,1
Расходы оборудование при работе web-приложения с учетом прочих расходов (2% от суммы)	7472,52

На эксплуатацию оборудования определим 12 месяцев (таблица 22).

Таблица 22 – Расчет капитальных затрат на создание и эксплуатацию системы web-приложения

Виды затрат	Единовременные капитальные затраты, руб.	Постоянные капитальные затраты (Кп)	
		время работы	сумма руб.
Затраты на разработку системы	36835,55		
Затраты на эксплуатацию оборудования		12 месяцев	7472,52
Всего	44308 руб.		

Рассчитаем трудовые затраты при базовом варианте для сотрудника Отдела ИО (таблица 23).

Таблица 23 – Расчет трудовых затрат при базовом варианте

Действие	Кол-во документов в год $Q_{д год}$ (шт)	Трудозатраты на один документ $T_{док}$ (чел/час)	Трудозатраты за год $T_{год}$ (чел/час)
Составление списка работников	12	0,5	6
Составление списка контрагентов (поставщиков)	12	0,5	6
Составление списка компьютерного оборудования	12	0,5	6
Составление списка комплектующих и расходников	12	0,5	6

Продолжение таблицы 23

Действие	Кол-во документов в год $Q_{д\ год}$ (шт)	Трудозатраты на один документ $T_{док}$ (чел/час)	Трудозатраты за год $T_{год}$ (чел/час)
Составление списка работ	12	0,5	6
Учет заявок	1095	1	1095
Составление документов на закупки	48	1	48
Составление документов по заявкам	1095	0,2	219
Печать документов по заявкам	1095	1	1095
Печать списков работников	12	0,5	6
Составление аналитических отчетов	12	1,5	18
Печать документов	12	0,5	6
Всего:			2517

Для расчета стоимостных затрат примем среднюю зарплату сотрудника Отдела ИО примерно 150 рублей в час.

При тестировании системы определено, что при автоматизированном варианте трудовые затраты снижаются примерно на 55%.

Все расчеты сведем в таблицу (таблица 24).

Таблица 24 – Стоимостные и трудовые затраты

Виды затрат	Базовый вариант	Новый вариант
Трудовые затраты	2517 чел/час	1132,65 чел/час
Стоимостные затраты	377550 руб.	169897,5 руб.

Общие затраты в год на разработку и эксплуатацию web-приложения

$$C = 44308 + 169897,5 = 214205,5 \text{ руб.}$$

Рассчитаем показатели эффективности, результаты сведем в таблицу 25.

Таблица 25 – Показатели эффективности

Показатель	Формула	Значение
Абсолютное снижение трудовых затрат	$\Delta T = T_0 - T_1$	1384,35 чел/час
Относительный коэффициент уменьшения трудовых затрат	$K_T$	55%
Индекс снижения трудовых затрат, повышение производительности труда	$Y_T = \frac{T_0}{T_1}$	1,8
Годовая экономия	$\Delta C = C_0 - C$	163344,5 руб.
Коэффициент экономической эффективности	$C_T = \left(\frac{\Delta C}{C}\right) \times 100\%$	76,3 %
Годовой экономический эффект	$\text{Эгод} = \Delta C - 76,3 \% \cdot 100 \cdot K$	135239 руб.
Период окупаемости проекта	$T_{\text{окуп.}} = \frac{K_{\text{п}}}{\Delta C}$	≈1,3 года

Капитальные затраты на разработку и реализацию web-приложения окупятся организации примерно в течение 15 месяцев, что свидетельствует об экономической эффективности инвестиций в разработку web-приложения. Срок внедрения такой системы невелик, а затраты на внедрение уже учтены в расчетах, поэтому срок окупаемости для web-приложения вполне приемлем.

Поэтому рассмотренный с экономической точки зрения проект эффективен.

Кроме того, развитие системы принесет косвенный социальный эффект: предлагаемая программа не только улучшит работу сотрудников Отдела защиты информации и развития информационных и коммуникационных систем по учету заявок на ремонт и модернизацию ПК, но и позволит повысить конкурентоспособность компании на рынке предоставления подобных услуг.

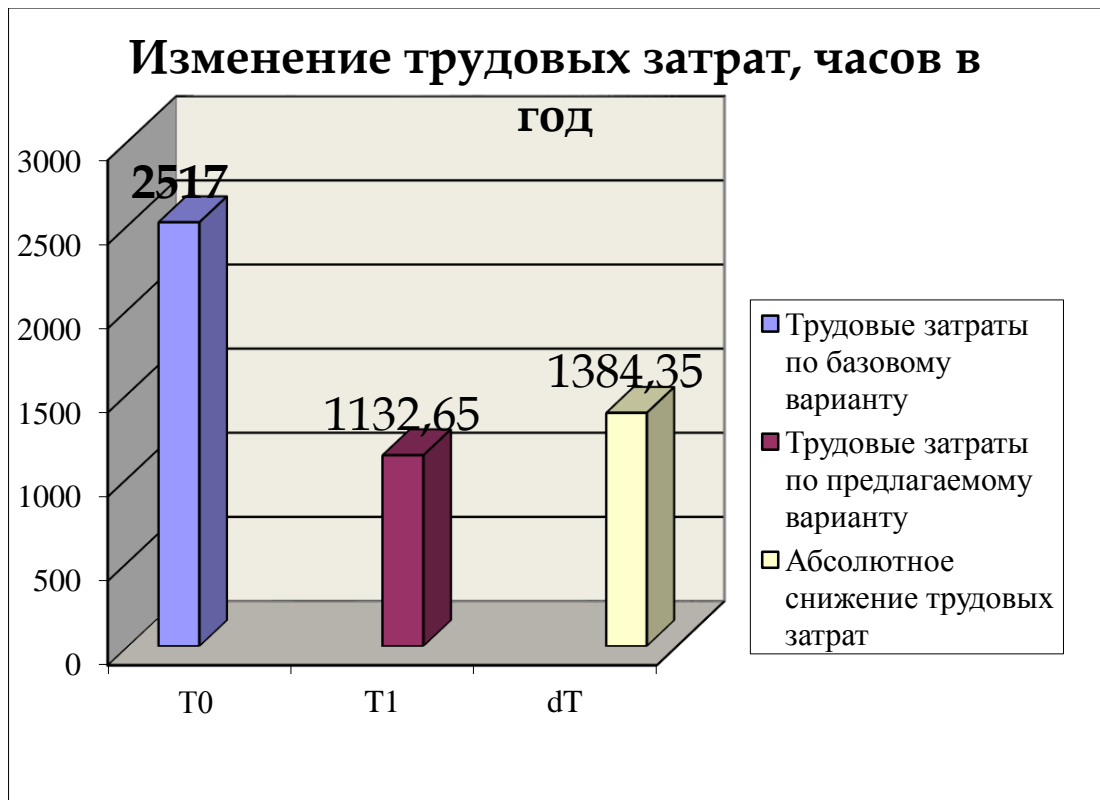


Рисунок 30 – Диаграмма снижения трудовых затрат

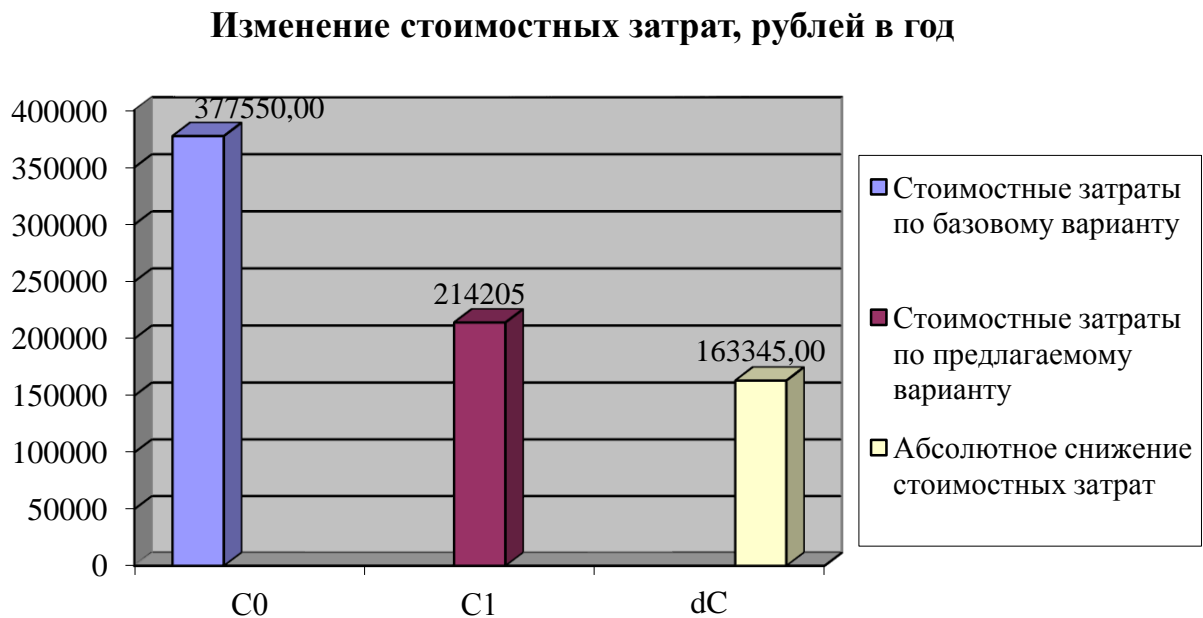


Рисунок 31 – Диаграмма снижения стоимостных затрат

Представленные диаграммы наглядно показывают эффективность

разработки web-приложения для учета заявок на техническую поддержку.

#### Выводы по 3 главе

Третья глава выпускной квалификационной работы посвящена расчетам и обоснованию экономической эффективности проекта.

Выбрана методика расчета экономической эффективности, согласно выбранной методике рассчитаны экономические показатели, выполнен их анализ, который показал эффективность разработанного решения.

## Заключение

В ходе выполнения бакалаврской работы проанализирована предметная область исследования, изучена организационная структура государственного автономного учреждения Иркутской области «Экспертиза в строительстве Иркутской области», (ГАУИО «Ирэкспертиза»), а также его отдела информационного обеспечения.

В результате выполненного исследования проанализированы бизнес-процессы техподдержки пользователей, выявлены недостатки и выполнен расчет временных характеристик существующей системы учета и после внедрения модуля.

Построены функциональные модели для описания существующих бизнес-процессов и бизнес-процессов техподдержки пользователей в условиях использования программного модуля.

На основании проведенного анализа определены требования к программному модулю техподдержки пользователей и рассмотрены существующие решения системы технической поддержки пользователей.

Выделены сущности предметной области и связи между ними, построена модель «Сущность-Связь», логическая модель данных и после анализа нормализации отношений между сущностями, построена модель физического представления базы данных.

Проанализированы информационные потоки и построена информационная модель процессов техподдержки пользователей в условиях использования программного модуля.

Рассмотрена схема клиент-серверного взаимодействия сайта и используемой СУБД, разработана функциональная структура программного модуля техподдержки с учётом клиент-серверной архитектуры, построена архитектура программного модуля техподдержки: его серверная и клиентская части.

Создана база данных, ее структура и наполнение.

После определения компонентов программного модуля построена его диаграмма компонентов.

При разработке интерфейса программного модуля техподдержки построено дерево функций, сценарий диалога и разработаны формы Web-интерфейса.

Все задачи, поставленные в выпускной квалификационной работе, выполнены, цель работы достигнута.

## Список используемой литературы и используемых источников

1. AllFusion ERwin Data Modeler 7.3 [Электронный ресурс]. – Режим доступа: <https://erwin.com/products/erwin-data-modeler/> (дата обращения 12.10.2022)
2. Claris – система оперативного управления бизнесом [Электронный ресурс]. – Режим доступа: WWW: <http://www.claris.su/> (дата обращения 11.10.2022)
3. Software by Web Help Desk. WWW [Электронный ресурс]. – Режим доступа: <http://www.webhelpdesk.com/> (дата обращения 11.10.2022)
4. Базы данных. Проектирование, реализация и сопровождение. Теория и практика, Томас Коннолли – М.: Вильямс, 2017. – 760с.
5. Веб-сайт php.net [Электронный ресурс]. - Режим доступа: <http://php.net/manual/ru/history.php.php> (дата обращения: 12.10.2022)
6. Веб-сайт SQL.ru [Электронный ресурс]. - Режим доступа: <http://www.sql.ru/> (дата обращения: 16.10.2022)
7. Вигерс Карл, Разработка требований к программному обеспечению, Пер, с англ. – М.:Издательско-торговый дом «Русская Редакция», 2004. – 576с.
8. Гагарина, Л.Г. Разработка и эксплуатация автоматизированных информационных систем. Учебное пособие / Л.Г. Гагарина. - М.: Форум, Инфра-М, 2015. - 384 с.
9. Гвоздева, Т. В. Проектирование информационных систем: технология автоматизированного проектирования. Лабораторный практикум : учебное пособие / Т. В. Гвоздева, Б. А. Баллод. – 2-е изд., стер. – Санкт-Петербург : Лань, 2020. – 156 с.
10. Денвер - локальный сервер [Электронный ресурс] Режим доступа: <http://www.denwer.ru/> (дата обращения: 14.10.2022)



- 11.Елизаров А.И. Технология разработки программного обеспечения: Учебное методическое пособие. / А.И.Елизаров, В.В. Романенко – Томск: Томский межвузовский центр дистанционного образования, 2007. – 119 с.
- 12.Золотов, С. Ю. Проектирование информационных систем: Учебное пособие [Электронный ресурс] / Золотов С. Ю. – Томск: ТУСУР, 2016. – 117 с. – Режим доступа: <https://edu.tusur.ru/publications/6478> (дата обращения: 12.10.2022)
- 13.Информатика: учеб. пособие / И.Н. Щапова, В.А. Щапов – Пермь: Изд-во Перм. нац. исслед. политехн. ун-та, 2016. – 154 с.
- 14.Коцюба И.Ю. Основы проектирования информационных систем. Учебное пособие. / И.Ю.Коцюба, А.В.Чунаев, А.Н.Шиков – СПб: Университет ИТМО, 2015. – 206 с.
- 15.Моделирование информационных систем : учебное пособие / В. В. Лисяк, Н. К. Лисяк ; Южный федеральный университет. – Ростов-на-Дону ; Таганрог : Издательство Южного федерального университета, 2018. – 88 с.
- 16.Остроух, А. В. Проектирование информационных систем : монография / А. В. Остроух, Н. Е. Суркова. – Санкт-Петербург : Лань, 2019. – 164 с.
- 17.Сайт Государственное автономное учреждение Иркутской области «Экспертиза в строительстве Иркутской области – [Электронный ресурс]. – Режим доступа: <http://irexpertiza.ru/> (дата обращения 10.10.2022)
- 18.Симдянов, И.В.; Кузнецов, М.В. MySQL 5; БХВ-Петербург - Москва, 2016. - 747 с.
- 19.Система LiveAgent Help Desk [Электронный ресурс]. – Режим доступа: <https://www.weebly.com/ec/app-center/liveagent?lang=ru> (дата обращения 11.10.2022)

20. Учебники, задачки, справочники по web языкам – [Электронный ресурс]. – Режим доступа: [http:// theory.phphtml.net](http://theory.phphtml.net) (дата обращения 14.10.2022)
21. Учебное пособие по Microsoft Visio – [Электронный ресурс]. – Режим доступа: <https://coderlessons.com/tutorials/microsoft-technologies/izuchite-microsoft-visio/uchebnoe-posobie-po-microsoft-visio> (дата обращения 10.10.2022)
22. Фаулер М. UML – основы. Руководство по стандартному языку объектного моделирования.: Пер. с англ. – СПб.: Символ, 2009. – 320 с.
23. Фаулер М. UML в кратком изложении: применение стандартного языка объектного моделирования: пер. с англ. / М. Фаулер, К. Скотт. – М.: Вильямс, 2010. – 580 с.
24. Федоров, Н.В. Проектирование информационных систем на основе современных CASE-технологий / Н.В. Федоров. – М.: МГИУ, 2018. – 280с.
25. Фрэйз, Б. HTML5 и CSS3. Разработка сайтов для любых браузеров и устройств / Б. Фрэйз ; [перевод с английского В. Черник]. – 2-е изд. – Санкт–Петербург [и др.] : Питер, 2017. – 512 с.
26. Цуканова О. А. Методология и инструментарий моделирования бизнес-процессов: учебное пособие – СПб.: Университет ИТМО, 2015. – 100 с.
27. Чистов, Д. В. Проектирование информационных систем. Учебник и практикум / Д. В. Чистов, П. П. Мельников, А. В. Золотарюк, Н. Б. Ничепорук. — М.: Юрайт, 2016. — 260 с.
28. Шаньгин В.Ф. Защита информации в компьютерных системах и сетях – М.: ДМК Пресс, 2012. - 592 с.
29. Этапы разработки баз данных [Электронный ресурс]: статья., Режим доступа <http://infis.narod.ru/db/db-10.htm>, свободный (дата обращения 11.10.2022).
30. Lurig М. PHP Reference: Beginner to Intermediate PHP5. Lulu.com, 2008. – 164 с.

31. Lynch, W., 2020, A Comprehensive Guide to 14 Types of UML Diagram, Available: <https://warren2lynch.medium.com/a-comprehensive-guide-to-14-types-of-uml-diagram-affcc688377e>.
32. PHP Programming Cookbook. Exelixis Media P.C., 2016. – 71 p.
33. Stevens P. Using UML: Software Engineering with Objects and Components. Addison-Wesley: Academic Service, 2006. – 280 p.
34. Watt, A. and Eng.N., 2014, Database Design – 2nd Edition. Victoria, B.C.: BCcampus. Retrieved from <https://opentextbc.ca/dbdesign01/>.

## Приложение А

### Экранные формы

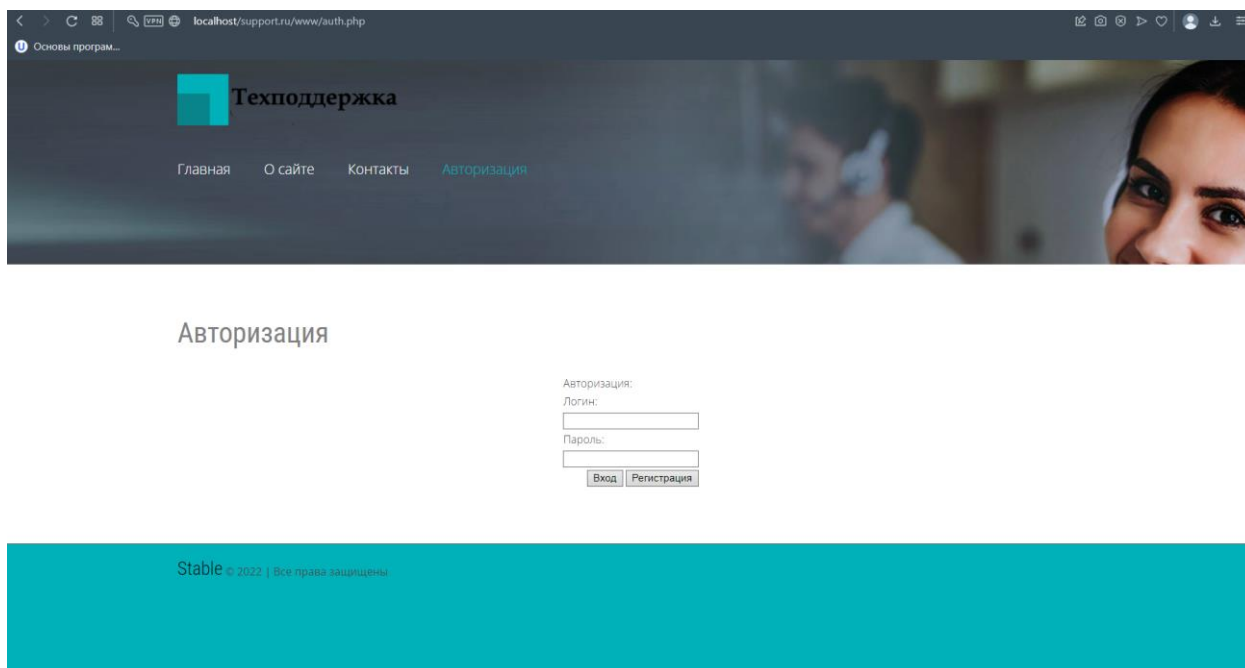


Рисунок А.1 –Форма авторизации

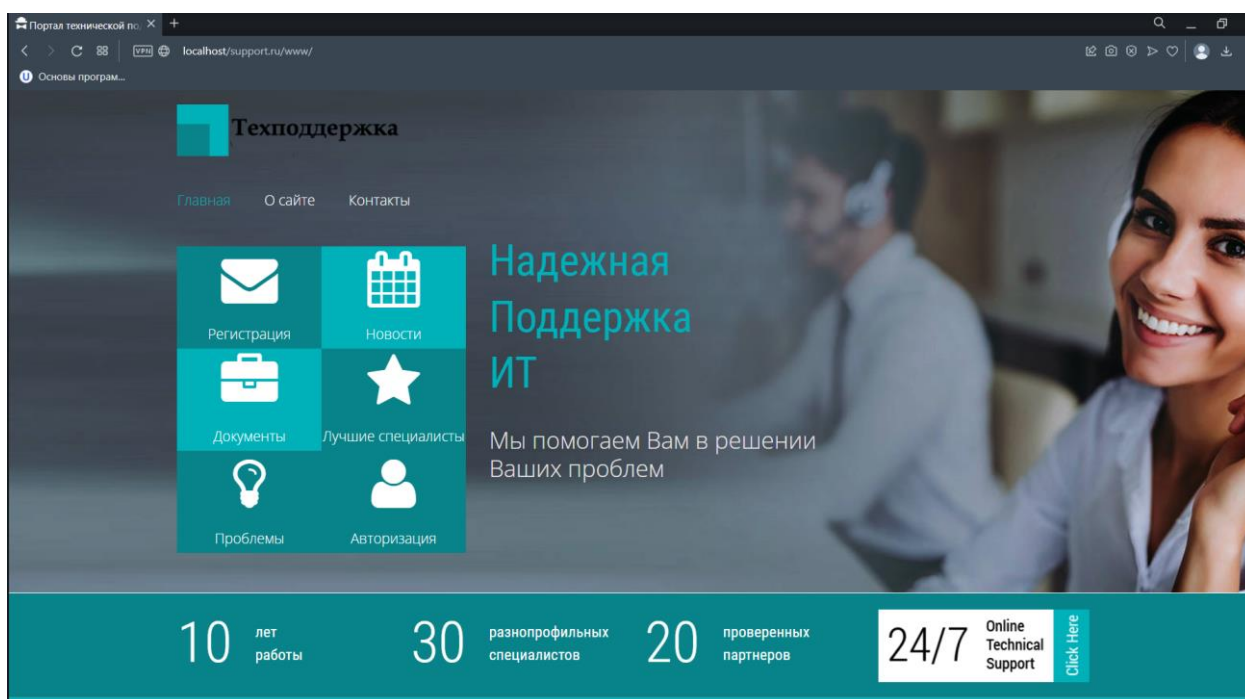


Рисунок А.2 – Главная форма системы

## Продолжение Приложения А

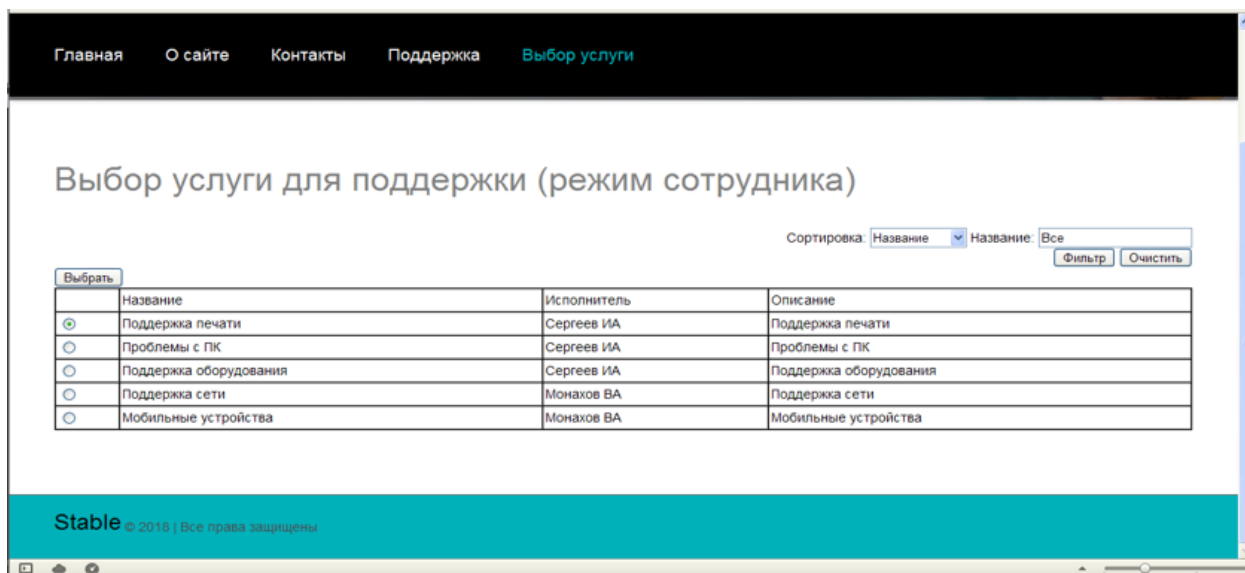


Рисунок А.3 – Форма страницы услуги для сотрудника

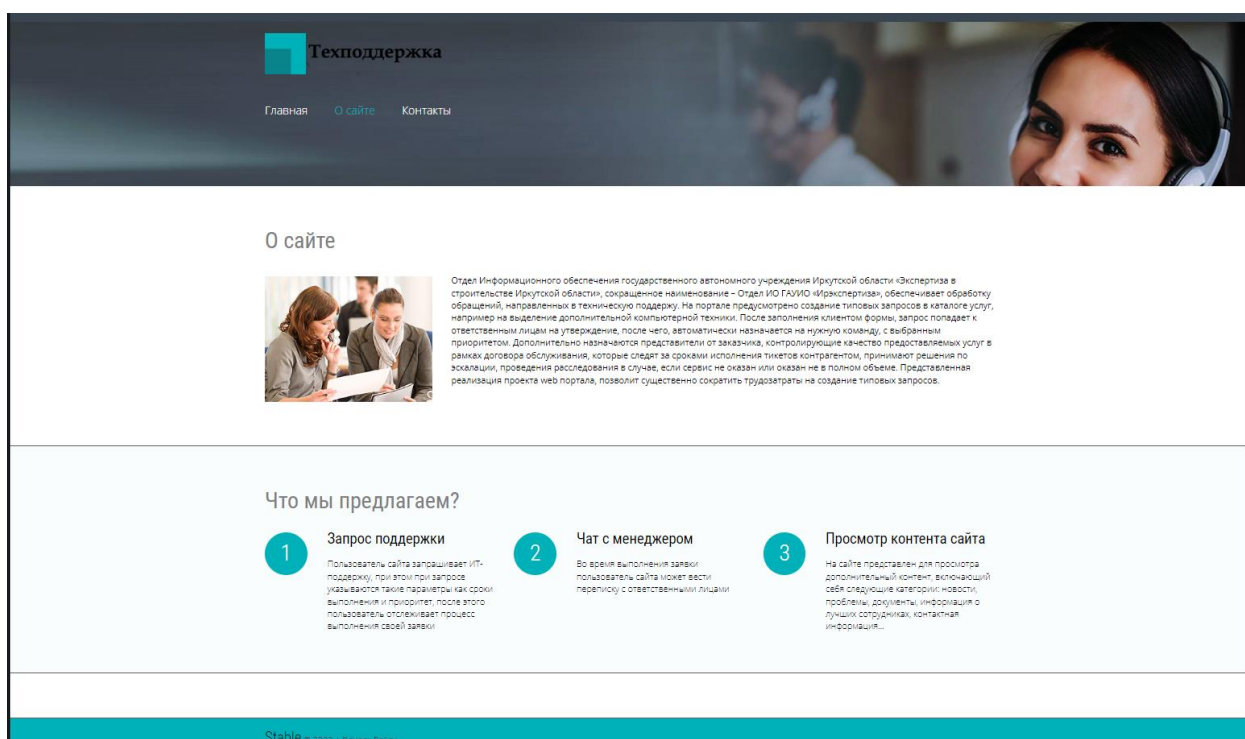


Рисунок А.4 – Форма страницы информации о сайте