

Министерство образования Кировской области

Кировское областное государственное профессиональное  
образовательное бюджетное учреждение  
«Кировский авиационный техникум»  
(КОГПОБУ «Кировский авиационный техникум»)

К защите ДП \_\_\_\_\_  
на заседании цикловой комиссии  
вычислительных специальностей  
Протокол № \_\_\_\_\_  
от «\_\_» \_\_\_\_\_ 2023 г.  
Председатель \_\_\_\_\_ О.А. Кононова

СОГЛАСОВАНО  
Заместитель директора  
по учебно-производственной  
работе  
\_\_\_\_\_ С.Г. Лубнин  
«\_\_» \_\_\_\_\_ 2023 г.

РАЗРАБОТКА ВЕБ-САЙТА ДЛЯ СБОРА  
ЗАЯВОК С ПРОБЛЕМНЫХ УЧАСТКОВ  
Пояснительная записка  
ДП.09.02.07.ИР41.013.ПЗ

Выполнил \_\_\_\_\_ А.М. Рамешк Асли

Руководитель ДП \_\_\_\_\_ М.Г. Чернядьева

Нормоконтроль  
пояснительной записки \_\_\_\_\_ Н.Ю. Мершина

Рецензент \_\_\_\_\_ О.А. Осмехина

Дипломный проект защищен на \_\_\_\_\_  
Протокол № 10 от «28» июня 2023 г.

# Содержание

Введение.....	3
1 Общая характеристика организации .....	6
2 Постановка задачи автоматизации .....	11
3 Проектные решения .....	15
3.1 Техническое обеспечение.....	15
3.2 Программное обеспечение .....	16
3.3 Информационное обеспечение .....	17
4 Технология разработки программного обеспечения.....	23
4.1 Общие сведения.....	23
4.2 Описание функциональной структуры .....	23
4.3 Руководство пользователя.....	25
4.4 Тестирование сайта .....	26
5 Экономическое обоснование разработки .....	28
6 Обеспечение безопасности сайта .....	37
Заключение .....	38
Приложение А (обязательное) Руководство пользователя.....	39
Приложение Б (обязательное) Тестирование вебсайта .....	52
Приложение В (обязательное) Техническое задание .....	54
Приложение Г(обязательное) Программный код вебсайта .....	67
Приложение Д (обязательное) Библиография.....	79

					ДП.09.02.07.ИР41.013.ПЗ					
Изм.	Лист	№ докум.	Подпись	Дата						
Разраб.		Рамешк Асли А.М.			Разработка веб-сайта для сбора заявок с проблемных участков Дипломный проект	Лит.	Лист	Листов		
Провер.		Чернядьева М.Г.						2	5	
Реценз.		Осмехина О.А.				КАТ, гр ИР-41				
Н. Контр.		Мершина Н.Ю.								
		.								

## Введение

Разработка веб-сайта для сбора заявок с проблемных участков имеет большую актуальность для ОАО «Коммунэнерго», поскольку в их ИТ отделе отсутствует эффективный механизм сбора заявок от клиентов. Наличие сайта, на котором пользователи могут заполнить форму заявки и взаимодействовать с представителями компании в режиме реального времени, значительно ускорит и упростит процесс сбора заявок и решения проблем.

Автоматизация процесса сбора заявок от клиентов является важным компонентом эффективного функционирования любого бизнеса. При подаче заявки клиент должен пройти регистрацию на сайте, чтобы получить доступ к форме заявки. Заполнение формы и отправка заявки должны быть удобными и простыми для клиента, чтобы минимизировать время и усилия, потребные на выполнение этого процесса.

После отправки заявки клиент получает уведомление о том, что его заявка успешно зарегистрирована в системе. Заявка имеет определенный статус, который отображает ее текущее состояние в системе.

Заявки, зарегистрированные в системе, автоматически отправляются в панель специалиста, ответственного за обработку заявок. Этот специалист может просматривать заявки, изменять их статусы и выполнять другие операции, связанные с обработкой заявок. Это помогает ускорить процесс обработки заявок, уменьшить количество ошибок и повысить удовлетворенность клиентов.

Важно, чтобы система автоматической обработки заявок была надежной и безопасной, чтобы защитить личную информацию клиентов и обеспечить защиту от мошенничества. Кроме того, эффективная система обработки заявок может предоставлять ценную аналитическую информацию, которая помогает бизнесу оптимизировать свои процессы и

					ДП.09.02.07.ИР41.013.ПЗ	Лист
Изм.	Лист	№ докум.	Подпись	Дата		3

улучшить качество обслуживания клиентов.

Цель - автоматизировать процесс сбора заявок от клиентов, ведения диалога с клиентами и сбора статистики о работе технической поддержки.

Задачи:

- Разработка функционала сбора заявок: создать механизм, позволяющий пользователям отправлять заявки через сайт. Это может включать в себя создание формы для заполнения заявки, ее обработку и хранение данных.
- Разработка функционала коммуникации с клиентами: предусмотреть возможность обратной связи с клиентами по электронной почте или через встроенный чат.
- Разработка панели управления: создать интерфейс, который позволит администраторам управлять заявками и обращениями клиентов.
- Разработка системы авторизации и аутентификации: создать систему, которая позволит пользователям и администраторам входить на сайт и получать доступ к соответствующим функциям.
- Оптимизация производительности и безопасности: убедиться, что сайт работает быстро и безопасно.
- Разработка дизайна и пользовательского интерфейса: создать удобный и интуитивно понятный интерфейс, который будет привлекать пользователей и упрощать работу с сайтом.
- Тестирование и отладка: провести тестирование сайта, чтобы выявить возможные ошибки и недочеты, а также отлаживать код для устранения ошибок и улучшения функционала.
- Документирование: создать документацию, которая поможет пользователям и администраторам понимать, как работать с сайтом и как использовать его функционал.
- Развертывание и поддержка: развернуть сайт на хостинге и обеспечить

					ДП.09.02.07.ИР41.013.ПЗ	Лист
Изм.	Лист	№ докум.	Подпись	Дата		4

его поддержку.

Ожидаемым результатом является создание бесплатного сервиса для сбора заявок с проблемных участков. Это поможет компании быстро общаться с клиентами, увеличить число пользователей услугами и повысить лояльность клиентов.

					ДП.09.02.07.ИР41.013.ПЗ	Лист
						5
Изм.	Лист	№ докум.	Подпись	Дата		

## 1 Общая характеристика организации

ОАО «Коммунэнерго» обеспечивает передачу электроэнергии потребителям и обслуживает линии наружного освещения в 45 населенных пунктах Кировской области, а в 5 районах вырабатывает и реализует населению и предприятиям тепловую энергию. По состоянию на 01.11.2022г. на балансе предприятия, а также на основании долгосрочных договоров на временное владение и пользование находится— 1588 трансформаторных подстанций и распределительных пунктов, 3592,2 км воздушных и 688,2 км кабельных линий электропередачи, 50 котельных. Подразделения компании работают во всех районах Кировской области [3].

Преддипломная практика была пройдена в ИТ-отделе организации ОАО «Коммунэнерго» (далее ОИТ). ОИТ находится по адресу: г. Киров, по адресу Солнечный проезд, 4. Дата регистрации организации - 13.09.2002.

Основными задачами ИТ-отдела являются:

- Формирование и развитие единой технической политики в области применения информационных технологий, систем обработки данных, средств телекоммуникации и связи, а также обеспечения информационной безопасности предприятия.
- Создание системы автоматизации управленческой, производственной и учетной деятельности предприятия на основе современных технологических решений с применением передовых методов оптимизации бизнес-процессов.
- Развитие информационной системы предприятия согласно изменяющимся внутренним потребностям, внедрение технологических платформ новых поколений.
- Обеспечение безотказной работы корпоративной вычислительной сети предприятия в круглосуточном режиме.

					ДП.09.02.07.ИР41.013.ПЗ	Лист
Изм.	Лист	№ докум.	Подпись	Дата		6

- Приведение эксплуатируемых автоматизированных учетных систем предприятия в соответствие происходящими изменениями действующего законодательства Российской Федерации.

Компания достигла высокого уровня развития информационных технологий, что позволяет обеспечить максимальную эффективность и автоматизацию всех бизнес-процессов.

Для эффективной работы предприятия необходимо внедрение автоматизации производственных процессов, что позволит повысить производительность и сократить время выполнения работ.

Автоматизация бизнес-процессов — это процесс применения современных информационных технологий для оптимизации, упрощения и автоматизации выполнения бизнес-процессов в организации. Целью автоматизации бизнес-процессов является улучшение эффективности работы предприятия, снижение затрат и повышение качества продукции или услуг [5].

Бизнес-процесс — это последовательность взаимосвязанных действий и операций, которые выполняются в предприятии или организации с целью достижения конкретного результата, удовлетворения потребностей клиентов и создания добавленной стоимости для компании [6]. Бизнес-процесс может включать в себя различные этапы, включая планирование, проектирование, исполнение, контроль и управление. Каждый бизнес-процесс может быть автоматизирован с помощью информационных технологий для оптимизации работы компании.

Автоматизация бизнес-процессов может включать в себя автоматизацию различных этапов процессов, включая сбор и обработку данных, контроль качества, управление производственными и финансовыми процессами и другие. Это может быть достигнуто с помощью различных программных и аппаратных средств, которые помогают управлять и контролировать бизнес-процессы на предприятии.

					ДП.09.02.07.ИР41.013.ПЗ	Лист
Изм.	Лист	№ докум.	Подпись	Дата		7

Задачи автоматизации бизнес-процессов:

- улучшение эффективности производства;
- оптимизация расходов;
- улучшение качества продукции;
- улучшение контроля над производством;
- увеличение скорости обработки информации;
- увеличение конкурентоспособности;
- повышение удовлетворенности клиентов.

ОАО «Коммунэнерго» имеет несколько информационных систем:

- ИС «КОММУНЭНЕРГО»;
- система программных решений компании 1С.

Организационная структура управления ОИТ представляет собой линейно-функциональную систему управления с разделением на директора и подразделения. Структурная схема ОИТ представлена на Рисунке 1.

Линейно-функциональная структура управления – это тип организационной структуры, в которой задачи разделены по функциям, а руководители отвечают только за свою функцию, что способствует более эффективному управлению предприятием. Преимущества структуры:

- эффективность работы;
- управление качеством услуг;
- эффективное использование ресурсов;
- эффективная координация между отделами.



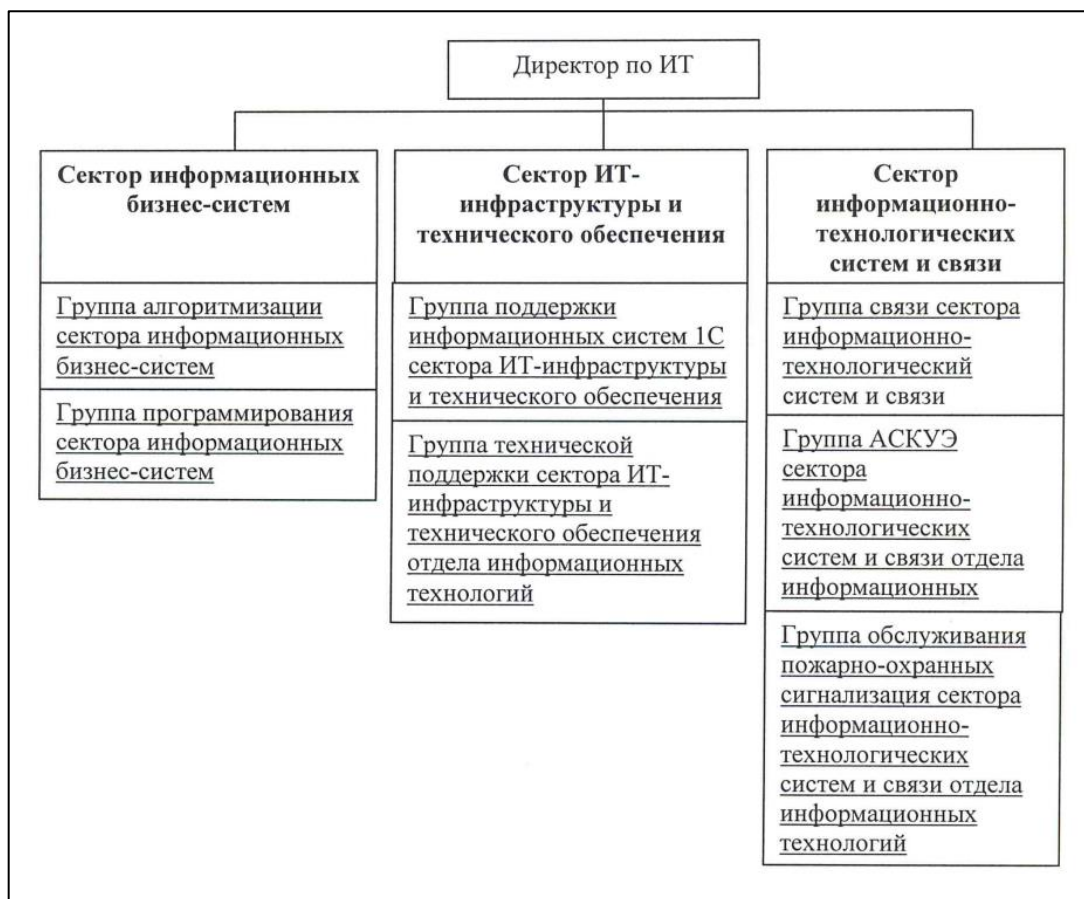


Рисунок 1 – Организационная структура ОИТ

Для обмена информацией между отделами и пользователями ОИТ используется локальная вычислительная сеть. Локальная вычислительная сеть предназначена для обеспечения совместной работы компьютеров внутри ИТ отдела. Она позволяет обмениваться информацией, ресурсами и управлять сетевыми устройствами в единой сетевой инфраструктуре. ЛВС также обеспечивает безопасность и защиту информации, позволяя настраивать доступ к различным ресурсам и контролировать передачу данных внутри сети. Это повышает эффективность работы ИТ отдела и улучшает коммуникацию между сотрудниками.

Топология вычислительной сети – звезда. В корпусах реализована совокупность технологий GigabitEthernet и FastEthernet (в качестве среды передачи используется неэкранированная витая пара категории 5е). Рабочие

станции в отделе подключаются к коммутатору, находящемуся на этаже расположения рабочего отдела. Коммутаторы отделов в свою очередь подключаются к главному коммутатору здания (мосту), расположенному на 1 этаже. Каждый находится в сетевом шкафу для ограничения доступа к ним посторонних лиц. Сеть построена на оборудовании фирм Allied Telesis, D-Link.

#### Характеристики рабочих компьютеров:

- процессор: AMD Ryzen 5 3600G (Socket AM4);
- оперативная память: Corsair Vengeance LPX 8GB DDR4 3000MHz (CMK8GX4M1D3000C16);
- жесткий диск: Seagate Barracuda 256GB 7200RPM 64MB Cache SATA 6.0Gb/s 3.5" (ST1000DM010);
- сетевая карта: TP-Link TG-3468 10/100/1000Mbps Gigabit PCI Express Network Adapter.

#### Характеристики серверов:

- тип процессора - два процессора Intel Xeon E5620 Quad-Core 2.4GHz;
- обновление процессора - поддержка до 2 процессоров;
- внутренняя кэш-память, Мб - кэш-память второго уровня 12 Мб;
- системная шина - "Шина QPI 5.86 GT/s";
- тип памяти - память PC3-10600R (DDR3-1333) и PC3-12800R (DDR3-1600);
- количество слотов оперативной памяти, шт. - 18 слотов DIMM;
- стандартный объём памяти, Гб - Стандартная память 4 Гб;
- внутренние дисковые отсеки - до 6 SAS/SATA с горячей заменой;
- внутреннее устройство хранения данных SATA - До 16 Тб;
- внутреннее устройство хранения данных SAS - До 12 Тб;
- в качестве серверной ОС установлен Ubuntu Server.

					ДП.09.02.07.ИР41.013.ПЗ	Лист
Изм.	Лист	№ докум.	Подпись	Дата		10

## 2 Постановка задачи автоматизации

Концепция создания модели работы для программного модуля связана с внедрением комплексных систем на производственных предприятиях. Этот подход предусматривает анализ и обследование бизнес-процессов предприятия с целью выявления потенциальных возможностей для оптимизации. На основе полученных данных разрабатывается модель работы вебсайта, которая должна соответствовать требованиям заказчика и позволять автоматизировать выявленные бизнес-процессы.

Создание такой модели работы вебсайта позволяет не только повысить эффективность работы предприятия, но и снизить затраты на ресурсы и увеличить конкурентоспособность компании на рынке. Для этого необходимо провести анализ рынка и выбрать оптимальное программное обеспечение, которое соответствует требованиям и возможностям компании. Функциональная модель изображена на Рисунке 2.

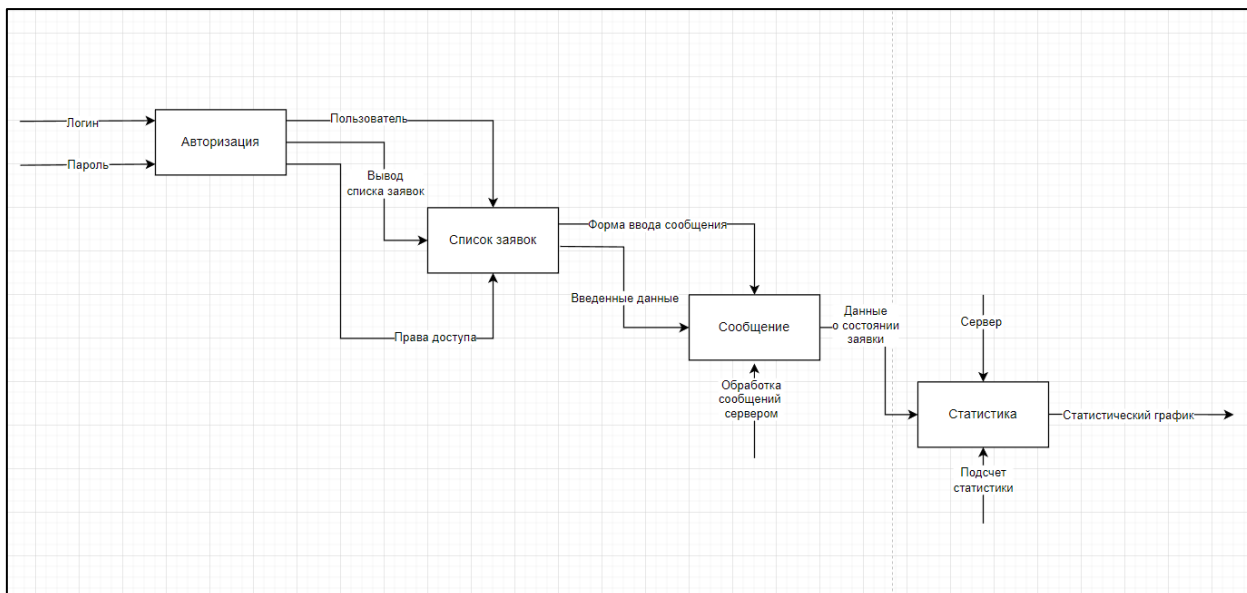


Рисунок 2 – Функциональная модель

Основные недостатки существующий практики управления и обработки информации:

- невозможность быстрой и эффективной обработки большого количества заявок;
- риск потери искажения информации при передаче данных по электронной почте;
- необходимость ручной обработки заявок, что увеличивает вероятность ошибок и задержек;
- ограниченный функционал для управления заявками и мониторинга их статуса;
- низкая прозрачность и невозможность отслеживания истории заявок;
- недостаточная автоматизация процессов, связанных с обработкой заявок, что увеличивает трудозатраты и время выполнения работы.

Веб-сайт поможет устранить перечисленные выше недостатки. Создание такого сайта направлено на улучшение следующий показателей:

- сокращение времени на обработку заявок за счет автоматизации процесса и упрощения процедуры подачи заявок;
- снижение количества ошибок при регистрации заявок благодаря обработке данных на стороне клиента и сервера;
- уменьшение затрат на обработку заявок и на обслуживание клиентов за счет уменьшения количества ручной работы и ускорения процессов;

Функции автоматизации при внедрении проекта:

- список зарегистрированных пользователей;
- список созданных ранее созданных заявок;
- создание новой заявки о проблеме;
- разграничение прав доступа;
- управление статусами заявок;
- управление назначением заявки;

					ДП.09.02.07.ИР41.013.ПЗ	Лист
Изм.	Лист	№ докум.	Подпись	Дата		12

– расчет данных о количестве заявок по фильтрам и графическое представление в виде графика.

– сбор данных о количестве заявок и выгрузка данных в xls или pdf.

Требования к проекту:

– авторизация пользователя и разграничение прав: сотрудник, клиент, администратор;

– поиск заявок по пользователю, который написал обращение и по тому, на кого назначена заявка, по дате создания заявки, по статусу и номеру заявки;

– возможность отправки сообщений от пользователя по созданной заявке;

– построение графика по заявкам с выбранным статусом и фильтрами даты, а также сотрудником, на которого назначена заявка;

– возможность выгрузить отчет в формате xls и pdf по статусам и датам заявок для всех сотрудников поддержки.

Порядок ввода первичной информации:

– Пользователь должен зарегистрироваться на сайте или авторизоваться, если он уже зарегистрирован.

– Пользователь нажимает кнопку «Создать заявку» и вводит тему обращения.

– Заявка обрабатывается на сервере и попадает в раздел «Заявки» к пользователям с доступом «Поддержка» или «Администратор».

– Сотрудник поддержки или администратор обрабатывают заявку, отвечают пользователю, анализируют статус решения проблемы и изменяют его при необходимости.

Характеристика результатов:

– После успешного создания заявки должно происходить перенаправление в диалог.

					ДП.09.02.07.ИР41.013.ПЗ	Лист
Изм.	Лист	№ докум.	Подпись	Дата		13

- Отображение отправленного в диалог сообщения и вложения.  
Изображения должны отображаться прямо в диалоге. Если вложение имеет другой формат, то отображается ссылка на скачивание.
- График, отображающий количество заявок с разным статусом по датам и сотрудникам, на которых назначена заявка.
- Таблицы в формате pdf или xls.

Обновление таблиц базы данных происходит, когда в базу данных вносится новая информация: новые отправленные сообщения, новые заявки, регистрация новых пользователей.

Используется диалоговой режим решения задачи. Пользователь вводит информацию в формы, далее информация обрабатывается на стороне сервера, пользователю представляется информация об успешном выполнении операции или об ошибке.

Один из примеров таких программ, carrotquest, предлагает функционал сбора заявок с сайта, обработки чата и e-mail, аналитики посетителей и взаимодействия с клиентами через социальные сети. Это позволяет компаниям собирать и анализировать информацию о поведении пользователей на сайте, предоставлять им персонализированный сервис и решать возникшие проблемы быстро и эффективно.

Другой продукт, Intercom, предоставляет возможности для общения с клиентами через чат, e-mail и SMS. Компания также предлагает аналитические инструменты для отслеживания поведения пользователей и управления коммуникациями с клиентами. Intercom предоставляет более продвинутые функции, такие как создание чат-ботов и автоматическое управление коммуникациями.

Оба конкурента предоставляют высокоэффективные инструменты для сбора заявок с сайта и общения с клиентами, но они платные. К примеру, в carrotquest тариф на месяц стоит 7990p/месяц, а в Intercome \$119.

					ДП.09.02.07.ИР41.013.ПЗ	Лист
Изм.	Лист	№ докум.	Подпись	Дата		14

### 3 Проектные решения

#### 3.1 Техническое обеспечение

Техническое обеспечение (ТО) — это совокупность материальных и программных компонентов, необходимых для создания, функционирования и обслуживания технических систем.

Техническое обеспечение может включать в себя компьютерное оборудование, сетевые устройства, программное обеспечение, технические средства связи, а также различные устройства, инструменты и материалы, необходимые для обслуживания и ремонта техники.

Состав технических средств:

Рекомендованные требования компьютера, с которого происходит доступ к сайту:

- процессор: 1,5 ГГц или выше;
- оперативная память: 2 Гб или выше;
- жесткий диск: 250 Гб или выше;
- видеокарта с поддержкой OpenGL 2.0 или выше;
- операционная система: windows 7 или выше, macOS x 10.7 или выше, linux с графической оболочкой.
- монитор: 14 дюймов или больше с разрешением 1024x768
- клавиатура и мышь.

Рекомендованные требования для сервера, на котором будут установлены веб-сайт и база данных:

- процессор: четырехъядерный процессор с тактовой частотой от 2,0 ГГц и выше;
- оперативная память: минимум 4 Гб оперативной памяти;
- хранилище: жесткий диск или твердотельный накопитель (SSD) емкостью от 100 Гб и более;

- сетевой интерфейс: Ethernet-адаптер с поддержкой скорости передачи данных от 1 Гбит/с и выше;
- операционная система: Linux Server 64bit;
- безопасность: сервер должен быть защищен от внешних атак (например, с помощью брандмауэра) и иметь механизмы резервного копирования данных для восстановления после сбоев.

Продукт будет использоваться на предприятии ОАО «Коммунэнерго».

Гарантийные условия действительны при соблюдении требований к техническому и программному обеспечению.

Полная совместимость с системами, поддерживающими графический интерфейс и современные браузеры.

Предполагаемый срок эксплуатации 3 года.

### 3.2 Программное обеспечение

Программное обеспечение - это комплекс программ, инструментов и данных, которые обеспечивают работу компьютерных систем и устройств. Оно состоит из компьютерных программ, инструкций и данных, которые позволяют компьютеру выполнять задачи, управлять оборудованием и взаимодействовать с пользователем [4].

Программное обеспечение может включать в себя операционные системы, прикладные программы, библиотеки, драйвера, среды разработки и многое другое.

Требования к программному обеспечению сервера:

- Python 3.1;
- Django Framework в виртуальной среде venv 4.2 и новее;
- Библиотеки openpyxl и reportlab для python;
- SqlLite;
- Docker.



### 3.3 Информационное обеспечение

Информационное обеспечение – это совокупность инструментов и методов, используемых для создания и управления информационной базой, которая включает в себя классификационные и кодировочные системы, унифицированные системы документации, схемы потоков информации, а также принципы и методы создания баз данных. Оно определяет способы и формы представления информации о состоянии объектов управления, как в виде данных внутри информационной системы, так и в виде документов, графиков и сигналов вне ее пределов.

Схема данных включает в себя описание содержания, структуры и ограниченной целостности, используемые для создания и поддержки базы данных.

Для хранения данных используется SQLite, так как сайт предполагает средне-низкий трафик и нагрузку на сервер.

Для использования веб-сайта создано 6 таблиц.

Таблица «auth\_user» хранит данные обо всех зарегистрированных на сайте пользователей, подробнее в Таблице 1. Имеет следующие поля:

- id (тип данных: integer) — идентификатор пользователя, автоматически генерируется при создании записи;
- password (тип данных: charfield) — пароль пользователя, хранится в зашифрованном виде;
- last\_login (тип данных: datetime) — дата и время последнего входа пользователя;
- username (тип данных: charfield) — логин пользователя, должен быть уникальным;
- first\_name (тип данных: charfield) — имя пользователя;
- last\_name (тип данных: charfield) — фамилия пользователя;

- email (тип данных: charfield) — адрес электронной почты пользователя;
- date\_joined (тип данных: datetime) — дата и время регистрации пользователя на сайте.

Таблица 1 – Структура таблицы «auth\_user»

Наименование полей	Тип данных	Ограничение	Ссылки	Пример
id (pk)	int	not null		5
username	charfield	50		Boris123
password	charfield	250		Fasdf87924!
first_name	charfield	50		Борис
last_name	charfield	50		Борисов
email	charfield	50		boris@mail.ru
last_login	datetime			12.12.2002 13:41
date_joined	datetime			12.12.2002 13:41

Таблица «auth\_group» — это таблица, для хранения данных о доступе пользователей, подробнее в Таблице 2. Имеет следующие поля:

- id (тип данных integer) — идентификатор группы пользователей;
- name (тип данных: charfield) – имя группы пользователей;
- permissions (тип данных ManyToMany) – поле, которое связывает группу с набором разрешений из таблицы "auth\_permission".

Таблица 2 – Структура таблицы «auth\_group»

Наименование полей	Тип данных	Ограничение	Ссылки	Пример
id (pk)	int	not null		5
name	charfield	50		Admin
permissions	ManyToMany			

Таблица «auth\_user\_groups» — это таблица, связывающая таблицы «auth\_group» и «auth\_users», подробнее в Таблице 3. Имеет следующие поля:

- id (тип данных: integer) — идентификатор записи;
- user\_id (тип данных: integer) — внешний ключ на таблицу пользователей (auth\_user);
- group\_id (тип данных: integer) — внешний ключ на таблицу групп (auth\_group).

Таблица 3 – Структура таблицы «auth\_user\_groups»

Наименование полей	Тип данных	Ограничение	Ссылки	Пример
id (pk)	int	not null		5
user_id (fk)	int		auth_user.id	5
group_id (fk)	int		auth_group.id	5

Таблица «conversation» отвечает за хранение обращений пользователей на сайте, подробнее в Таблице 4. Поля таблицы и их типы данных:

- id (тип данных: integer) — идентификатор обращения;
- user1 (тип данных: integer) — внешний ключ на таблицу «auth\_user» (ForeignKey), связь многие-к-одному, поле отвечает за отправившего заявку пользователя;
- user2 (тип данных: integer) — внешний ключ на таблицу «auth\_user» (ForeignKey), связь многие-к-одному, поле отвечает за сотрудника поддержки;
- title (тип данных: charfield) — описывает тему обращения;

- created\_at (тип данных: datetime) — автоматически заполняется, текущей датой и временем при создании записи в таблице
- status (тип данных: charfield) — поле с ограниченным выбором значений (choices), заданных в списке STATUS\_CHOICES, по умолчанию установлено значение 'active'.

Таблица 4 – Структура таблицы «conversations»

Наименование полей	Тип данных	Ограничение	Ссылки	Пример
id (pk)	int	not null		1
user 1 (fk)	int	not null	auth_users. id	6
user2 (fk)	int	not null	auth_users. id	5
title	charfield			Здравствуйте, почему такой большой счет за коммунальные услуги в этом месяце?
status	charfield			Активна
created_at	datetime			12.12.2002 13:41

Таблица "message" предназначена для хранения сообщений пользователей и содержит следующие поля:

- id (тип данных: integer) — уникальный идентификатор сообщения (автоинкремент).
- text (тип данных: textfield) — текст сообщения.
- conversation (тип данных: integer) — внешний ключ, связывающий сообщение с конкретным диалогом, указывающий на таблицу "conversation".

- sender (тип данных: integer) — внешний ключ, связывающий сообщение с конкретным отправителем, указывающий на таблицу "user\_auth".
- created\_at (тип данных: datetime) — дата и время создания сообщения (автоматически генерируются).

Дополнительные подробности представлены в Таблице 5.

Таблица 5 – Структура таблицы «message»

Наименование полей	Тип данных	Ограничение	Ссылки	Пример
id (pk)	int	not null		1
text	textfield			Да, здравствуйте!
conversation (fk)	int		conversation.id	5
sender	charfield			boris123
created_at	datetime			12.12.2002 13:41

Таблица "attachment" используется для хранения вложений, которые пользователь прикрепляет. Она содержит следующие поля:

- message (тип данных: integer) — внешний ключ, связывающий модель с таблицей "message". При удалении записи из таблицы "message", все связанные с ней записи из таблицы "attachment" также будут удалены из базы данных.
- file (тип данных: filefield) — поле, в котором сохраняется файл, выбранный пользователем.
- uploaded\_at (тип данных: datetime) — поле, в котором хранится дата и время загрузки файла.

Дополнительные подробности представлены в таблице 6.

Таблица 6 – Структура таблицы «attachment»

Наименование полей	Тип данных	Ограничение	Ссылки	Пример
id (pk)	int	not null		1
message_id(fk)	int		message.id	2
file	filefield			image.png
uploaded_at	datetime			12.12.2002 13:41

ER-диаграмма (Entity-Relationship diagram, диаграмма сущность-связь) — это графическое представление структуры базы данных, которое позволяет описать сущности (объекты), их атрибуты и связи между ними.

ER-диаграмма изображена на Рисунке 3.

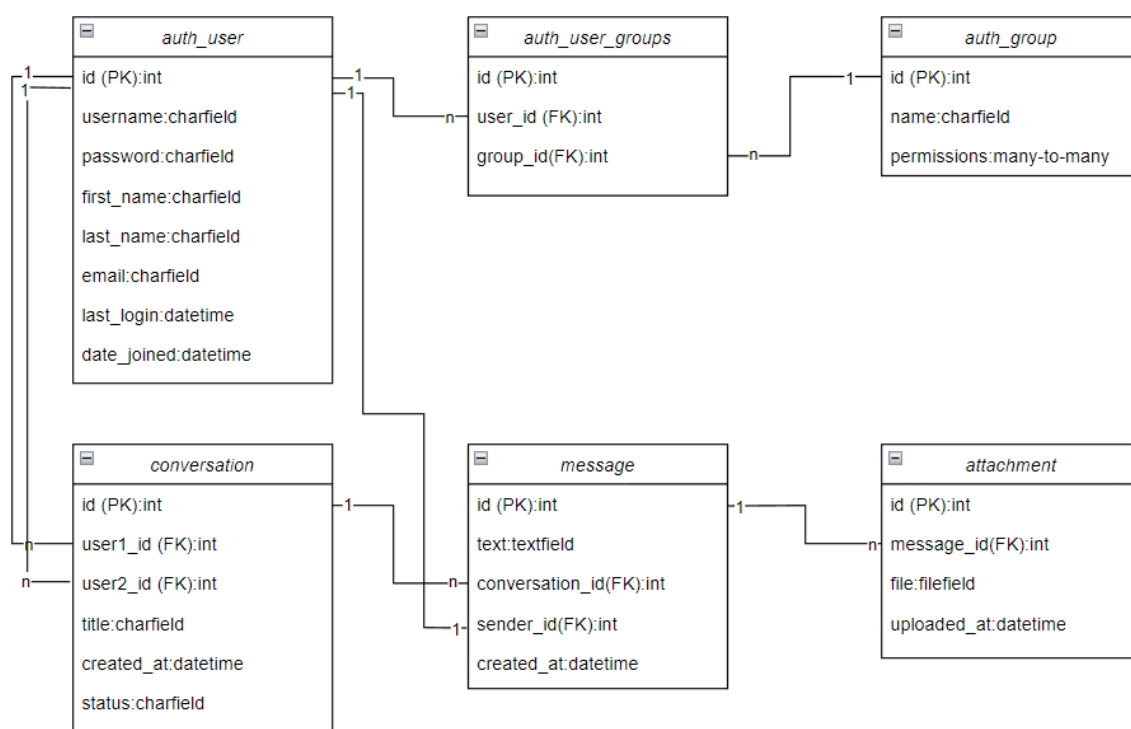


Рисунок 3 – ER-диаграмма

## 4 Технология разработки программного обеспечения

### 4.1 Общие сведения

Наименование темы разработки – «Веб-сайт для сбора заявок с проблемных участков».

Программное обеспечение, необходимое для функционирования программы:

- Google Chrome 113.0.5672.92 или любой другой современный браузер;
- операционная система с графическим интерфейсом, поддерживающая работу браузера.

Для разработки вебсайта была использована: среда разработки: PyCharm 2023.1, в качестве веб фреймворка использовалась библиотека Django языка Python, БД – SqlLite, СУБД – панель администрирования Django.

Классы решаемых задач:

- анализ данных;
- визуальное представление информации;
- оптимизация управления;

Объем исходных текстов программы 7.74 Мб.

### 4.2 Описание функциональной структуры

Вебсайт имеет один способ вызова – переход по url адресу <http://cr54935.tw1.ru/>. Возможно добавление другого домена в панели управления хостинга.

Перечень выполняемых функций:

- разграничение прав доступа;

					ДП.09.02.07.ИР41.013.ПЗ	Лист
Изм.	Лист	№ докум.	Подпись	Дата		23

- построение графика, отображающего количество заявок с разным статусом по датам и сотрудникам, на которых назначена заявка.
- возможность выгрузить отчет в формате xls и pdf по статусам и датам заявок для всех сотрудников поддержки;
- создание новой заявки;
- поиск заявок по дате создания, отправителю, статусу, номеру заявки, сотруднику, на которого назначена заявка;
- отображение всех ранее написанных по заявке сообщений;
- отправка нового сообщения;
- прикрепление вложения к сообщению;
- возможность удаления и корректирования всех сущностей через панель управления Django.

Выходные данные:

- страница регистрации: сообщение об ошибке ввода данных для всех полей формы;
- страница авторизации: сообщение об ошибке ввода данных в поля логина и пароля;
- страница «Статистика»: статистический график за выбранный период по заявкам с выбранными статусами. По умолчанию берется статистика за последний месяц. В ней содержатся форматы: число, дата, текст, график.
- таблицы в формате pdf, либо xls;
- страница «Заявки»: кнопки: Создать новую заявку, Фильтры, Фильтровать (button); поля для ввода текста: Номер заявки, Дата обращения, Статус заявки, Отправитель, Назначен на сотрудника (input type="text"); таблица списка всех заявок со следующими столбцами: Номер заявки, Тема обращения, Отправитель, Назначен на сотрудника, Статус (table).



- форма диалога: блоки сообщений: справа блок авторизованного пользователя, слева блок того, кто отвечает (блоки div); поле для ввода сообщения (компонент textarea), кнопка Отправить (button), блок со сменой статуса заявки и сотрудника, на которого назначена заявка (компоненты select).
- страница «Пользователи»: таблица со всеми зарегистрированными пользователями (table).

### 4.3 Руководство пользователя

Руководство пользователя — это документ, который описывает, как использовать программное обеспечение, оборудование или систему. Это может быть написанный документ, веб-страница, видеоуроки, интерактивные обучающие модули и другие формы документации. Руководство пользователя обычно включает в себя инструкции по установке, настройке и использованию продукта, а также описание функций и возможностей, которые доступны пользователю.

Руководство пользователя ГОСТ 19.505–79 рассматривается как документ, в котором указаны конкретные действия оператора. Основная задача оператора – осуществлять обслуживание системы или программного обеспечения входящего в систему, поэтому руководство пользователя частично объединяет в себе информацию, предназначенную для оператора и программиста системы.

Структура руководства пользователя:

- назначение программы;
- условия выполнения программы;
- выполнение программы;
- сообщения пользователю.

Руководство пользователя представлено в Приложении А.

#### 4.4 Тестирование сайта

Тестирование программного обеспечения — это процесс проверки программного продукта на соответствие требованиям и ожиданиям заказчика, выявление дефектов и ошибок, а также оценка его качества. Цель тестирования программного обеспечения состоит в том, чтобы обеспечить достаточную уверенность в том, что продукт работает корректно, соответствует требованиям и ожиданиям пользователей, а также готов к выпуску на рынок.

Тестовые наборы и результаты представлены в Приложении Б.

Функциональное тестирование — это тип тестирования программного обеспечения, в котором проверяются функциональные требования продукта. Оно направлено на проверку того, что программное обеспечение работает так, как ожидалось, и выполняет свои функции в соответствии с требованиями заказчика.

В функциональном тестировании проверяется, соответствует ли программа функциональным требованиям, как она обрабатывает входные данные, взаимодействует с пользователем, реагирует на ошибки и исключительные ситуации.

Преимущества тестирования:

- функциональное тестирование полностью имитирует фактическое использование системы ПО.
- тестирование помогает выявить системные ошибки ПО, предотвращая проблемы при его дальнейшей работе.
- исправление ошибок на более раннем этапе жизненного цикла ПО экономит ресурсы.

					ДП.09.02.07.ИР41.013.ПЗ	Лист
Изм.	Лист	№ докум.	Подпись	Дата		26

Поскольку сотрудники, использующие сайт, могут иметь разный уровень подготовки в области ИТ, необходимо протестировать интерфейс.

Проверка эргономичности позволяет определить, удобен ли искусственный объект (например, веб-страница, пользовательский интерфейс или устройство) для его предполагаемого применения. Проверка эргономичности измеряет эргономичность объекта или системы.

Золотое сечение — это пропорциональное деление отрезка на неравные части, при котором меньший отрезок относится к большему так же, как большая часть относится к меньшей.

Кошелек Миллера - емкость памяти ограничена семью цифрами, буквами или названиями предметов.

Принцип группировки предполагает разделение экрана программы на ясно очерченные блоки элементов, возможно, с заголовком для каждого блока. Принцип бритвы

Оккама или KISS в дизайне интерфейсов подразумевает минимальное число действий для выполнения любой задачи, логичную последовательность действий и оптимизацию движения курсора и глаз пользователя.

Видимость отражает важность выноса на первый план самой важной информации и элементов управления, сделав их доступными для пользователя. Менее важную информацию можно переместить в меню.

Принцип умного заимствования позволяет использовать широко распространенные приемы дизайна интерфейсов и удачные находки авторов конкурирующих программ для сокращения времени обучения и повышения комфорта пользователя.

## 5 Экономическое обоснование разработки

Экономический анализ является важным инструментом для повышения эффективности деятельности организаций и укрепления их финансового состояния. Он представляет собой научную дисциплину, которая исследует экономику организаций с целью оценки их работы по достижению бизнес-планов, оценки финансового состояния и выявления потенциальных резервов для повышения эффективности.

Для разработчика программного обеспечения, как и для любого другого производителя, главная задача заключается в том, чтобы окупить затраты на разработку и получить прибыль. В данном контексте существуют два варианта:

- производитель находит заказчика, который оплачивает разработку программного обеспечения;
- разработка окупает себя будущими продажами программного обеспечения.

До начала проектирования сайта необходимо четко сформулировать цели и задачи, и на основе них определить примерный перечень работ по созданию сайта, состав команды разработчиков и функции, выполняемые каждым членом команды.

Исходя из вышеуказанных задач состав команды разработки представлен в Таблице 7.

Таблица 7 – Состав команды разработки вебсайта

Должность	Численность (чел.)	Тарифный разряд	МРОТ	Тарифный коэффициент	Месячн ый оклад, руб
Программист	1	4	13 890	1,6	22 224

Необходимо определиться с перечнем работ по созданию вебсайта, рассчитать время выполнения этих задач разработчиком, а также трудоемкость выполнения каждой задачи.

Трудоемкость выполнения работ рассчитывается по формуле

$$T_p = \frac{3t_{\min} + 2t_{\max}}{5}, \quad (1)$$

где  $t_p$  – расчетная трудоемкость выполнения работы(час),

$t_{\min}$  – минимальное время, необходимое для выполнения работы,

$t_{\max}$  – максимальное время, необходимое для выполнения работы.

Перечень работ по созданию вебсайта, а также время участия каждого разработчика приведены в Таблице 8.

Таблица 8 – Трудоемкость выполнения работ

Наименование работы	$t_{\min}$	$t_{\max}$	$t_p$	Программист
Изучение задания	16	24	19,2	18
Подбор и изучение литературы и патентов	16	24	19,2	20
Анализ проблемы и существующих алгоритмов.	24	40	30,4	31
Разработка общих принципов построения вебсайта и представления данных.	24	56	36,8	42
Финансово–экономическое обоснование создания вебсайта. Проведение маркетинговых исследований.	24	40	30,4	28
Выбор и обоснование СУБД, операционной системы, инструментария. Разработка структуры вебсайта	40	80	56	60
Разработка новых алгоритмов.	40	80	56	70
Разработка пользовательского интерфейса.	32	56	41,6	42
Разработка входных и выходных форм.	24	32	27,2	22

Наименование работы	tmin	tmax	tp	Программист
Установка файловой системы сайта на хостинг	8	16	11,2	10
Отладка, тестирование, корректировка, устранение выявленных ошибок	16	40	25,6	24
Разработка документации.	16	40	25,6	20
Написание инструкции для пользователя на русском языке.	10	16	12,4	8
Печать документации.	1	8	3,8	1
Всего	291	552	395,4	416

Например, написание инструкции пользователя на русском языке будет равно:

$$Tp = \frac{3 * 10 + 2 * 16}{5} = 12,4$$

Все работы группируются по комплексам. Комплексы работ по разработке вебсайта для ОАО «Коммунэнерго» приведены в Таблице 9.

Таблица 9 – Комплексы работ по созданию автоматизированной информационной систем

Наименование комплекса работ	Обозначение	Тр	Программист
1. Проектирование сайта и написание его кода.	Вмо	105,6	104
2. Установка сайта на хостинг, тестирование, отладка.	Вм	11,2	10
3. Прочие затраты по разработке	Впр	211,2	220
4. Маркетинговые исследования	Вми	0	0
5. Оформление программного продукта	Воф	67,4	53

Наименование комплекса работ	Обозначение	Тр	Программист
Всего:	Впо	395,4	387

Затраты – выраженные в денежной форме издержки, возникшие в процессе производства товаров и оказания услуг в целях получения прибыли. Они включают в себя:

Затраты на разработку можно определить по формуле:

$$З_{рп} = З_{мо} + З_{ком} + З_{пр}, \quad (2)$$

где  $З_{мо}$  – затраты на создание математического обеспечения и написание кода сайта;

$З_{ком}$  – затраты, связанные с работой компьютера при разработке вебсайта;

$З_{пр}$  – прочие затраты, связанные с разработкой вебсайта.

Затраты на создание математического обеспечения и написание программы определяется по формуле:

$$З_{мо} = З_{Пмо} + В_{рк} + С_{сн} * ФОТ_{мо} + Н_{р}, \quad (3)$$

где  $З_{Пмо}$  – затраты на выплату заработной платы работникам, участвующим в создании математического обеспечения и написании кода вебсайта;

$С_{сн}$  – общая ставка отчислений на социальные нужды;

$ФОТ_{мо}$  – фонд оплаты труда работников, участвующих в создании вебсайта;

$Н_{р}$  – накладные расходы организации, где разрабатывается вебсайт.

Затраты на выплату заработной платы работников определяется по формуле

$$ЗП_{мо} = ЗП_{п} + ЗП_{др}, \quad (4)$$

где ЗП<sub>п</sub> – заработная плата программистов, участвующих в создании математического обеспечения и написании кода вебсайта;

ЗП<sub>др</sub> – заработная плата других работников временной творческой группы, принимающих участие в разработке математического обеспечения и написании кода вебсайта. В данном случае это руководитель и оператор.

Заработная плата определяется по формуле

$$ЗП_{п} = P * В_{мо} * \frac{О_{м}}{ds * Д_{р}}, \quad (5)$$

где Р – число работников определенной квалификации, участвующих в разработке математического обеспечения и написании кода вебсайта;

В<sub>мо</sub> – время участия работника определенной квалификации в разработке математического обеспечения и написании кода вебсайта.

Заработная плата программиста, рассчитанная по формуле, будет равна:

Зарплата программиста:

$$ЗП_{др} = 1 * 104 * \frac{22\,224}{8 * 22} = 13\,541,23 \text{ (руб.)}$$

Премия этим работникам определяется по формуле

$$П = \frac{ЗП_{мо} * 20\%}{100\%}, \quad (6)$$



Премия равна:

$$П = \frac{13\,541,23 * 20\%}{100\%} = 2\,708,25 \text{ (руб.)}$$

Выплаты по районному коэффициенту определяется по формуле

$$Врк = \frac{(ЗПмо + П) * 15\%}{100\%}, \quad (7)$$

$$Врк = \frac{(13\,541,23 + 2\,708,25) * 15\%}{100\%} = 2\,437,42 \text{ (руб.)}$$

Фонд оплаты труда работника, участвующего в создании математического обеспечения и написании кода вебсайта, можно найти по формуле

$$ФОТмо = ЗПмо + П + Врк, \quad (8)$$

ФОТ работника, создающего математическое обеспечение вебсайта равен:

$$ФОТмо = 13\,541,23 + 2\,708,25 + 2\,437,42 = 18\,686,90 \text{ (руб.)}$$

Определив накладные расходы организации, можно рассчитать затраты на создание математического обеспечения и написание кода вебсайта. Накладные расходы организации показаны в Таблице 10.

					ДП.09.02.07.ИР41.013.ПЗ	Лист
Изм.	Лист	№ докум.	Подпись	Дата		33

Таблица 10 – Накладные расходы

Наименование	Ед. изм.	Цена, руб.	Количество	Сумма, руб.
Электроэнергия	кВт	4,27	220	939,40
Всего:				939,40

Затраты на создание математического обеспечения и написании кода вебсайта, рассчитанные по формуле, будут равны:

$$З_{мо} = 18\,686,90 + 30\% * 18\,686,90 + 939,40 = 25\,232,37(\text{руб.})$$

Рассчитаем общий фонд оплаты труда работника, премии, выплаты по районному коэффициенту, ФОТ специалиста участвующего в создании вебсайта:

Оператор:

$$\text{Зарплата: } З_{Пр} = 1 * 104 * \frac{22\,224}{8*22} = 13\,541,23;$$

$$\text{Премия: } П = \frac{13\,541,23 * 20\%}{100\%} = 2\,708,25;$$

Выплаты по районному коэффициенту:

$$В_{рк} = \frac{(13\,132,36 + 2\,626,47) * 15\%}{100\%} = 2\,437,42(\text{руб.})$$

ФОТ:

$$ФОТ = 13\,541,23 + 2\,708,25 + 2\,437,42 = 18\,686,90 (\text{руб.})$$

Для наглядности оформим расчет затрат на создание вебсайта в Таблицу 11.

					ДП.09.02.07.ИР41.013.ПЗ	Лист
Изм.	Лист	№ докум.	Подпись	Дата		34

Таблица 11 – Смета затрат на создание вебсайта

Наименование статьи затрат	Буквенное обозначение	Сумма, руб.
Зарплата программистов	ЗПпр	13 541,23
Премия	П	2 708,25
Выплата по районному коэффициенту	Врк	2 437,42
Накладные расходы	Нр	939,40
Затраты, связанные с работой компьютера при разработке прикладного решения	Зком	2,40
Затрат на разработку прикладного решения	Зрп	32 228,13

Расчет прибыли.

Прибыль (Приб) рассчитываем по формуле

$$\text{Приб} = \text{Зпр} - \text{ФОТ}, \quad (9)$$

$$\text{Приб} = 32\,228,13 - 13\,541,23 = 18\,686,90 (\text{руб.})$$

Расчет цены реализации программного продукта

Цену реализации (Цреал) вебсайта, с учетом налога на добавленную стоимость (НДС = 20%).

$$\text{Цреал} = \text{Цпр} + \text{НДС}, \quad (10)$$

$$\text{Цреал} = 32\,228,13 + 32\,228,13 * 20\% = 38\,673,75 (\text{руб.})$$

Расчет рентабельности (Р) от продажи вебсайта

Рентабельность вебсайта от его продажи по формуле

$$R = (\text{Приб} / \text{ФОТ}) * 100\%, \quad (11)$$

$$P = (18\,686,90 / 38\,673,75) * 100\% = 48,32\%$$

Расчет срока окупаемости (Ток) программного продукта по формуле

$$Tок = Приб / ФОТ, \quad (12)$$

$$Tок = 38\,673,75 / 18\,686,90 = 2,1 \text{ месяца}$$

Создание вебсайта для компании целесообразно и для заказчика. Это заключается в следующем:

- затраты, связанные с созданием вебсайта гораздо меньше, чем ручная обработка заявок несколькими специалистами, а также содержание горячей телефонной линии.
- заказчику выгодно приобрести такой вебсайт, так как примерный срок окупаемости составляет 2,1 месяца.

Приобретение и использование веб-сайта для компании является выгодным с экономической точки зрения, поскольку разработка вебсайта имеет очень низкий срок окупаемости.

Вебсайт обеспечивает существенную экономию денежных средств путем автоматизации задач и отказа от ручного выполнения, что в свою очередь освобождает часть финансовых ресурсов.

## 6 Обеспечение безопасности сайта

Базовый принцип защиты данных в клиент – серверном варианте заключается в том, что пользователи не имеют прямого доступа к файлам информационной базы.

При входе в информационную базу пользователь указывает свой логин и пароль. Если в системе существует пользователь с соответствующими правами, то доступ разрешается.

Для обеспечения защиты от несанкционированного доступа и целостности:

- на сайте присутствует авторизация;
- пароли в базе данных хранятся в зашифрованном виде;
- разграничение прав пользователей.

					ДП.09.02.07.ИР41.013.ПЗ	Лист
Изм.	Лист	№ докум.	Подпись	Дата		37

## Заключение

В рамках дипломного проекта был разработан веб-сайт для сбора заявок с проблемных участков, полностью соответствующий заявленным требованиям, а также руководство пользователя, которое в полном объеме описывает работу с сайтом, а также включает в себя описание основных страниц веб-сайта и функции.

В процессе разработки была проанализирована текущая система сбора заявок, а также сервисы-аналоги, предоставляющие подобные услуги. На основе этих данных были разработаны требования к разрабатываемому сайту.

В процессе разработки использовался следующий стек технологий: Python, Django Web Framework, HTML, CSS, JavaScript, jQuery. Программный код был написан в среде разработки PyCharm.

Объектом автоматизации стал отдел технической поддержки компании ОАО «Коммунэнерго».

Предмет автоматизации – сбор заявок с проблемных участков.

Разработка сайта с проблемных участков может принести множество преимуществ, включая повышение удобства использования для пользователей, улучшение SEO-оптимизации и повышение общей эффективности сайта, что может привести к увеличению посещаемости и конверсии. Кроме того, устранение проблемных участков может снизить количество отказов и увеличить время, проведенное пользователем на сайте, что способствует повышению уровня удовлетворенности и повышению лояльности клиентов.

Можно сделать вывод о том, что все задачи решены, цель достигнута.

					ДП.09.02.07.ИР41.013.ПЗ	Лист
Изм.	Лист	№ докум.	Подпись	Дата		38

Приложение А  
(обязательное)  
Руководство пользователя

РАЗРАБОТКА ВЕБ-САЙТА ДЛЯ СБОРА ЗАЯВОК С  
ПРОБЛЕМНЫХ УЧАСТКОВ

Руководство пользователя

ДП.09.02.07.ИР41.013

Листов 12

					ДП.09.02.07.ИР41.013.ПЗ	Лист
Изм.	Лист	№ докум.	Подпись	Дата		39

## Аннотация

В документе приведено руководство для специалиста технической поддержки по эксплуатации «Сайт для сбора заявок с проблемных участков».

В разделе «Назначения вебсайта» приведена информация о функциях вебсайта.

В разделе «Условия работы сайта» указаны требования к техническому и программному обеспечению.

В разделе «Работа с вебсайтом» описана последовательность действий сотрудник технической поддержки, которые обеспечат выполнение его обязанностей, а также перечень реакций вебсайта на эти действия.

В разделе «Сообщения пользователю» приведены текста сообщений, которые будут выводиться пользователю в ходе работы с сайтом.

					ДП.09.02.07.ИР41.013.ПЗ	Лист
Изм.	Лист	№ докум.	Подпись	Дата		40



## Содержание

1	Назначение вебсайта .....	42
2	Условия работы вебсайта .....	43
3	Работа с вебсайтом .....	44
4	Сообщения пользователю .....	51

					ДП.09.02.07.ИР41.013.ПЗ	Лист
Изм.	Лист	№ докум.	Подпись	Дата		41

## 1 Назначение вебсайта

В процессе выполнения дипломного проекта был разработан вебсайт для сбора заявок с проблемных участков.

Функции вебсайта:

- возможность отправить сообщение со стороны клиента и со стороны сотрудника технической поддержки;
- управление статусами заявок;
- разграничение прав доступа;
- создание графика, отражающего деятельность сотрудника технической поддержки, отображение написанных ранее сообщений;
- возможность выгрузить отчет в формате xsl и pdf по статусам и датам заявок для всех сотрудников поддержки;
- поиск заявок и пользователей по фильтрам.

					ДП.09.02.07.ИР41.013.ПЗ	Лист
Изм.	Лист	№ докум.	Подпись	Дата		42

## 2 Условия работы вебсайта

Рекомендованные требования к программному обеспечению:

- операционная система: windows 7 или выше, macos x 10.7 или выше, linux с графической оболочкой;
- любой современный браузер, поддерживающий html5 и js.

Рекомендованные требования к техническому обеспечению:

- процессор: 1,5 ггц или выше;
- оперативная память: 2 гб или выше;
- жесткий диск: 250 гб или выше;
- видеокарта с поддержкой opengl 2.0 или выше;
- операционная система: windows 7 или выше, macos x 10.7 или выше, linux с графической оболочкой;
- монитор: 14 дюймов или больше с разрешением 1024x768;
- клавиатура и мышь.

					ДП.09.02.07.ИР41.013.ПЗ	Лист
Изм.	Лист	№ докум.	Подпись	Дата		43

### 3 Работа с вебсайтом

Для работы с сайтом нужно перейти по адресу <http://cr54935.tw1.ru/>. После перехода по адресу произойдет перенаправление на страницу авторизации. Для авторизации необходимо ввести логин и пароль в соответствующие поля. Процесс авторизации изображен на Рисунке 1.

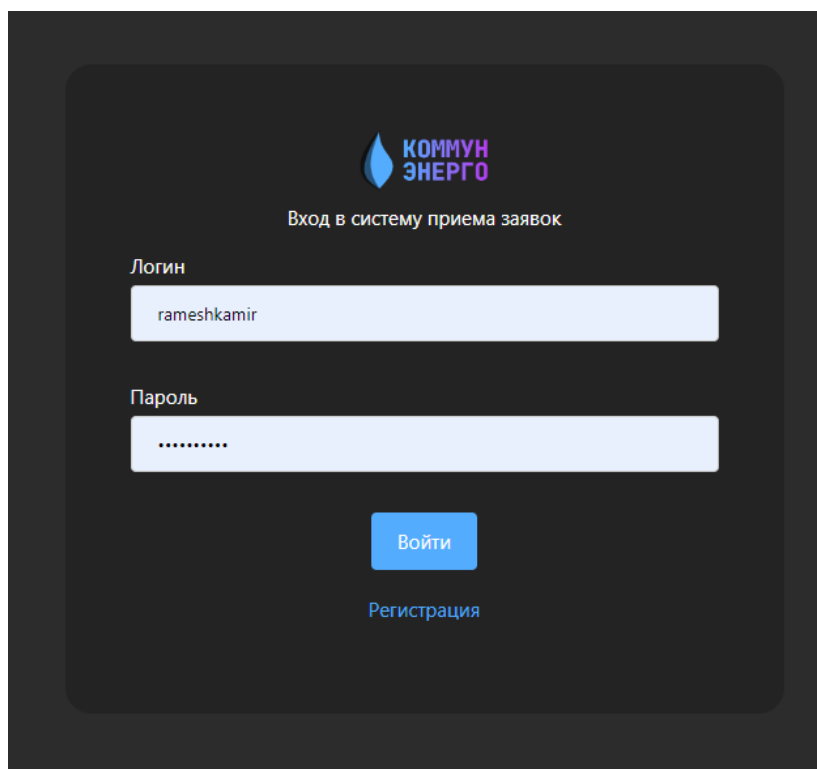

The image shows a login interface on a dark background. At the top center is a logo consisting of a blue flame icon and the text 'КОММУН ЭНЕРГО' in blue and purple. Below the logo is the text 'Вход в систему приема заявок'. There are two input fields: the first is labeled 'Логин' and contains the text 'gameshkamir'; the second is labeled 'Пароль' and contains a series of dots. Below the password field is a blue button with the text 'Войти'. At the bottom of the form area is a blue link labeled 'Регистрация'.

Рисунок 1 – Процесс авторизации

Если у пользователя нет аккаунта, то необходимо нажать на ссылку «Регистрация». Произойдет перенаправление на страницу регистрации. Для регистрации необходимо заполнить поля: логин, электронная почта, имя, фамилия, пароль и повторный ввод пароля. Процесс авторизации изображен на Рисунке 2.

  
 Регистрация

Логин

Электронная почта

Имя

Фамилия

Пароль

Подтвердите пароль

Зарегистрироваться

Войдите, если есть аккаунт

Рисунок 2 – Процесс регистрации

После регистрации пользователь получает доступ «Обычные пользователи» (далее обычные пользователи) и попадает на страницу «Заявки». Если же пользователь с доступом «Поддержка» или «Администратор» (далее сотрудники) авторизовался на сайте, то он попадает на страницу «Статистика». Для обычных пользователей на странице «Заявки» отображаются следующие элементы:

- Кнопка «Создать новую заявку» - после нажатия появляется поле для ввода темы обращения, а также кнопку «Создать заявку».
- Текстовое поле «Тема обращения» - поле для ввода темы обращения.
- Кнопка «Создать заявку» - создает заявку с введенной темой.
- Кнопка «Фильтры» - отображает/скрывает фильтры.
- Текстовое поле «Номер заявки» - фильтр по номеру заявки.
- Поле с выбором даты «Дата обращения» - фильтр по дате создания заявки.
- Кнопка «Фильтровать» - фильтрует заявки по заданным фильтрам.

Для сотрудников на странице «Заявки» отображаются все вышеперечисленные элементы, а также следующие элементы:

- Поле выбора «Статус заявки» - фильтр по статусу заявки.
- Текстовое поле «Отправитель» - фильтр по пользователю, который отправил обращение.
- Текстовое поле «Назначен на сотрудника» - фильтр по сотруднику, на которого назначена заявка.

Страница «Заявки» содержит таблицу с заявками, которые были найдены в соответствии с уровнем доступа и фильтрами.

Обычным пользователям отображаются только заявки, которые они отправляли. Таблица для обычных пользователей содержит столбцы: «Номер заявки», «Тема обращения», «Дата создания».

Сотрудникам отображаются все заявки, и таблица содержит такие же столбцы, как и у пользователей с доступом «Обычные пользователи», а также столбцы: «Отправитель», «Назначен на сотрудника», «Статус».

Страница «Заявки» для обычных пользователей изображена на Рисунке 3.

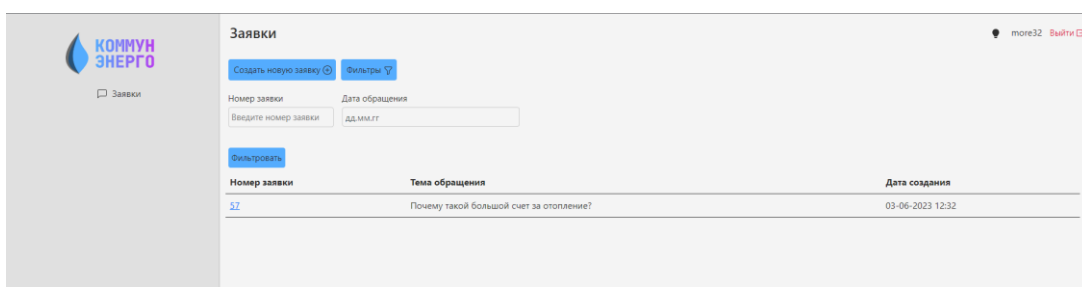


Рисунок 3 – Страница «Заявки» для обычных пользователей

Страница «Заявки» для сотрудников изображена на Рисунке 4.

Номер заявки	Тема обращения	Дата создания	Отправитель	Назначен на сотрудника	Статус
42	F	15-05-2023 18:11	Амир Рамешк	Амир Рамешк	Активная
47	123	23-05-2023 15:52	Амир Рамешк	Кирилл Коробов	Закрыта
55	5	25-05-2023 00:32	Амир Рамешк	Амир Рамешк	Закрыта
49	У меня проблема	23-05-2023 19:09	Валентин Валентинов	Амир Рамешк	Закрыта
43	asdf	23-05-2023 15:27	Амир Рамешк		Отложена
44	фыва	23-05-2023 15:37	Амир Рамешк		Активная
45	ыва	23-05-2023 15:47	Амир Рамешк		Активная
46	123	23-05-2023 15:52	Амир Рамешк		Отложена
48	a	23-05-2023 17:19	Амир Рамешк		Активная

Рисунок 4 – Страница «Заявки» для сотрудников

После создания новой заявки происходит переадресация на страницу диалога с поддержкой. Если заявка уже была создана, то на страницу диалога можно перейти, нажав на номер заявки.

На странице диалога для обычных пользователей отображаются следующие элементы:

- Заголовок «Тема обращения» - отображается тема обращения.
- Окно с чатом – в нем отображается диалог с технической поддержкой.
- Поле для ввода сообщения.
- Кнопка в виде бумажного самолетика – отправка набранного сообщения.
- Кнопка в виде скрепки – открытие окна с выбором вложения для прикрепления.
- Блок с сообщением – содержит в себе: текст сообщения, дату отправки, имя отправителя, прикрепленное изображение или ссылку на скачивание прикрепленного вложения.

Для сотрудников отображаются те же элементы, что и для обычных пользователей, а также:

- Поле с выбором «Статус заявки» - переключение статуса заявки.
- Поле с выбором сотрудника – переключение сотрудника, на которого назначена заявка.
- Поле с именем и фамилией клиента, который отправил заявку.

Страница с диалогом для обычных пользователей изображена на Рисунке 5.

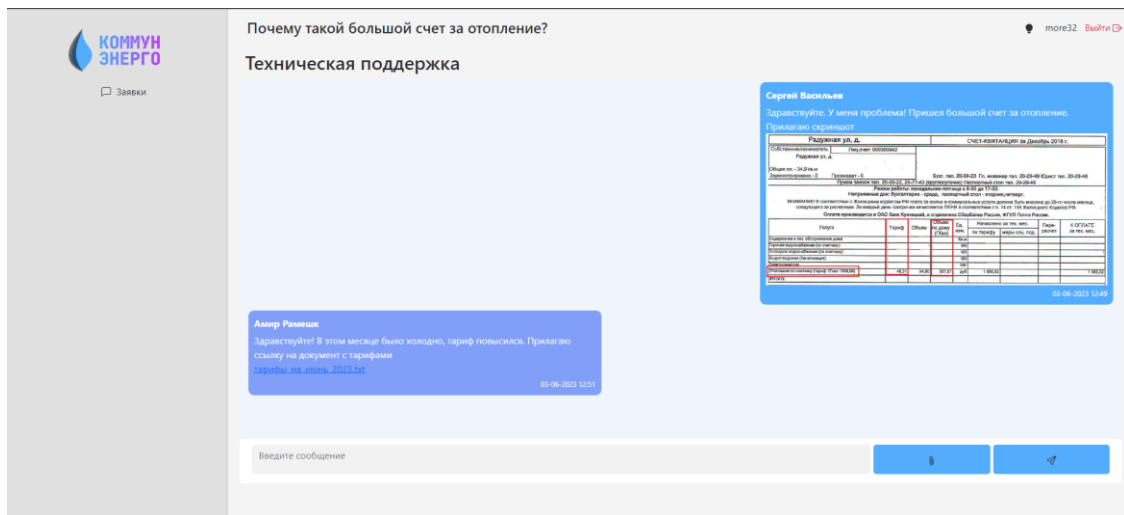


Рисунок 5 - Страница с диалогом для обычных пользователей

Страница с диалогом для сотрудников изображена на Рисунке 6.

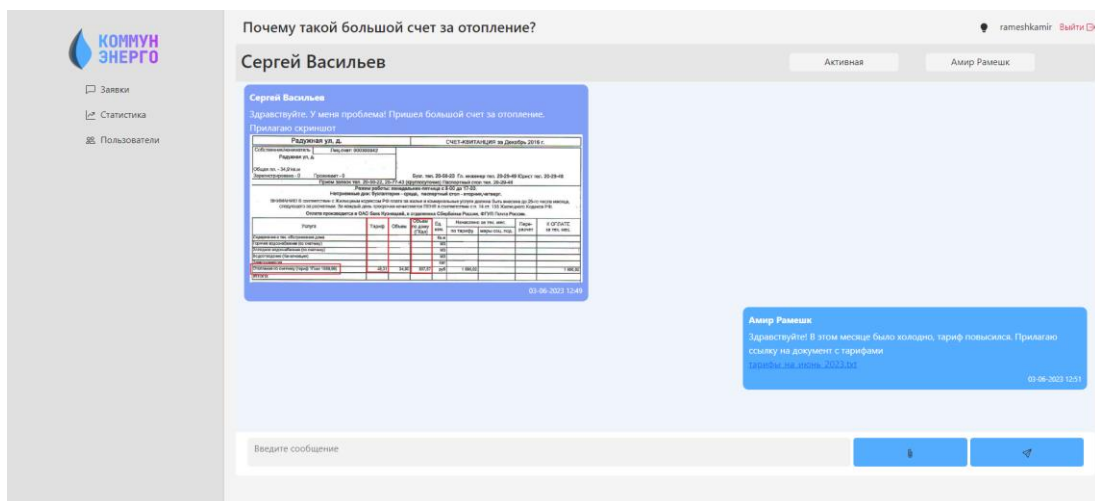


Рисунок 6 - Страница с диалогом для сотрудников

Для сотрудников также доступны еще две страницы: «Статистика» и «Пользователи».

Страница «Пользователи» отображает данные о зарегистрированных пользователях и содержит следующие элементы:

- Кнопка «Фильтры» - скрывает/отображает фильтры.
- Текстовое поле «Имя пользователя» - фильтр по имени пользователя.
- Текстовое поле «Уровень доступа» - фильтр по уровню доступа пользователя.



- Кнопка «Фильтровать» - применяет заданные фильтры.
- Таблица, отображающая всех зарегистрированных пользователей. Таблица содержит следующие столбцы: имя, фамилия, уровень доступа, обращения (этот столбик содержит все обращения пользователя с ссылками на диалоги) и дата последнего входа.

Страница «Пользователи» отображена на Рисунке 7.

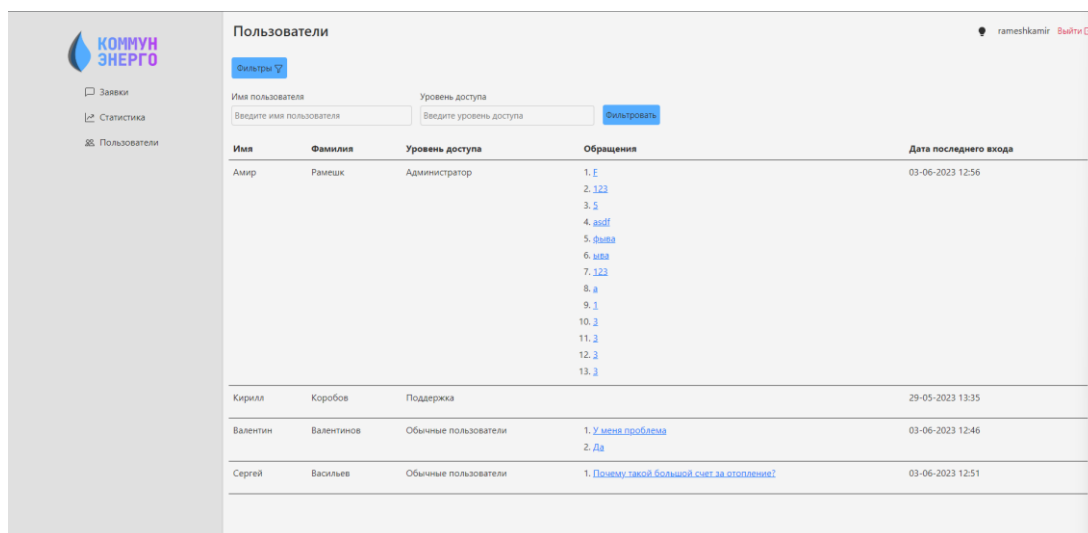


Рисунок 7 – страница «Пользователи»

На странице «Статистика» происходит отрисовка графика, который отображает работу технической поддержки в соответствии с заданными фильтрами. На этой странице имеются следующие элементы:

- Кнопка «Фильтры» - отображает/скрывает фильтры;
- Поле с выбором даты «Заявки от» - отвечает за выбор даты создания заявки, с которой нужно начать выбор заявок для отрисовки графика.
- Поле с выбором даты «Заявки до» - отвечает за выбор даты создания заявки, до которой нужно сделать выбор заявок для отрисовки графика.
- Кнопка «Фильтровать» - применяет выбранные фильтры и перерисовывает график.
- Поле с графиком – по оси y отображено количество заявок, по оси x отображены даты поступления заявок, в легенде содержатся переключатели для выбора статуса заявок.

Для сотрудников с доступом «Администратор» доступна статистика всех сотрудников с доступами «Поддержка» или «Администратор», тогда как сотруднику с доступом «Поддержка» доступна только собственная статистика. Для доступа «Администратор» на страницу «Статистика» добавляются следующие поля:

- текстовое поле «Назначен на сотрудника», необходимое для выбора сотрудника, по которому необходимо посмотреть статистику;
- кнопка «PDF» - выгружает отчет по заявкам в формате pdf;
- кнопка «EXCEL» выгружает отчет по заявкам в формате xls.

Страница «Статистика» отображена на Рисунке 8.

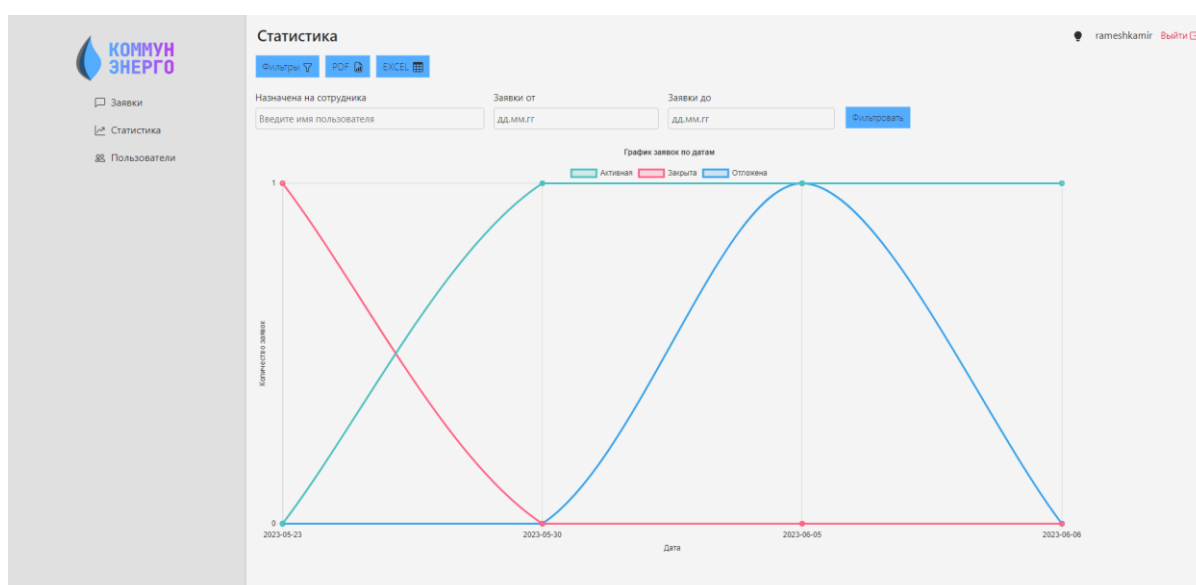


Рисунок 8 – Страница «Статистика»

На сайте присутствует навигация в левом меню. Каждый пункт перенаправляет на страницу с соответствующим названием и назначением. Также есть кнопка смены темы в правом верхнем углу в виде лампочки и кнопка выхода из учетной записи «Выйти».

#### 4 Сообщения пользователю

При вводе уже указанных другими пользователями имени пользователя или почты на форме регистрации выдается ошибка «Пользователь с таким именем уже существует».

При вводе некорректных данных в фильтры выводятся ошибки «Введите число» или «Введите дату».

При отправке сообщения в чат с диалогом добавляется новый блок с сообщением, с параметрами, который ввел пользователь.

					ДП.09.02.07.ИР41.013.ПЗ	Лист
Изм.	Лист	№ докум.	Подпись	Дата		51

Приложение Б  
(обязательное)  
Тестирование вебсайта

Таблица 12 – Функциональное тестирование вебсайта

Действия	Предполагаемый результат	Выполнение действия	Проверка программы
Ввод уже указанных другими пользователями имени пользователя или почты на форме регистрации	Сообщение о том, что эти данные уже используются другими пользователями	Вывод сообщения «Пользователь с таким именем/почтой уже существует.»	Тест пройден
Некорректный ввод данных в поля фильтров	Сообщение о некорректном заполнении данных	Вывод сообщения «Введите число», «Введите корректную дату»	Тест пройден
Отправка вложения без сообщения	Сообщение о невозможности отправить вложение без сообщения	Вывод сообщения «Введите сообщение»	Тест пройден
Ввод текста в поля для ввода чисел	Ввод символов ограничен для этих полей	Возможно ввести только числа	Тест пройден

Таблица 13 – Тестирование интерфейса вебсайта

Принцип	Обеспечение	Результат
Кошелек Миллера	Страница «Заявки»: <ul style="list-style-type: none"> <li>– кнопка создания заявки;</li> <li>– фильтры;</li> <li>– кнопка скрытия фильтров.</li> <li>– кнопка фильтрация</li> <li>– таблица с выводом заявок;</li> </ul> содержит не больше 9 элементов.	0,6

Принцип	Обеспечение	Результат
	Страница «Статистика» – график; – фильтры; – кнопка фильтровать; содержит не больше 7 кнопок	0,9
	Страница «Пользователи»: –фильтры; –кнопка фильтровать; –таблица с выводом всех пользователей; содержит не более 7 элементов	0,6
Принцип группировки	Весь сайт разбит на две части: левая – навигация и правая – рабочая область.	0,5
Бритва Оккама или KISS	Кнопка «Фильтровать» - фильтрует записи по заданным фильтрам	0,7
	Кнопка «Создать заявку» - открывает форму для создания новой заявки	0,6
	Кнопка «Фильтры» - скрывает/показывает фильтры	0,7
	Кнопка в виде скрепки – прикрепление вложения, кнопка в виде самолетика – отправка сообщения	0,8

Приложение В  
(обязательное)  
Техническое задание

РАЗРАБОТКА ВЕБ-САЙТА ДЛЯ СБОРА ЗАЯВОК С  
ПРОБЛЕМНЫХ УЧАСТКОВ

Техническое задание

ДП.09.02.07.ИР41.013.ТЗ

Листов 13

					ДП.09.02.07.ИР41.13.ПЗ	Лист
						54
Изм.	Лист	№ докум.	Подпись	Дата		

## Содержание

Введение.....	56
1 Основания для разработки .....	57
2 Назначение разработки.....	58
2.1 Функциональное назначение .....	58
2.2 Эксплуатационное назначение .....	58
3 Требование к программе или программному изделию .....	59
3.1 Требования к функциональным характеристикам .....	59
3.2 Требования к надежности .....	59
3.3 Условия эксплуатации .....	60
3.4 Требования к составу и параметрам технических средств.....	61
3.5 Требования к информационной и программной совместимости.....	61
3.6 Специальные требования .....	62
4 Требования к программной документации .....	63
5 Техничко-экономические показатели .....	64
6 Стадии и этапы разработки .....	65
7 Порядок контроля и приемки .....	66

## Введение

Наименование – «Веб-сайт для сбора заявок с проблемных участков». Веб-сайт предназначена для ОАО «Коммунэнерго».

Веб-сайт представляет собой систему сбора заявок от пользователей, которые пользуются услугами организации. Заявку может подать любой пользователь интернета, имеющий адрес электронной почты. Важность разработки сайта для сбора заявок обусловлена тем, что на почту приходит достаточно много заявок, которые обрабатывать становится всё труднее и труднее. Такой сайт позволит автоматизировать процесс приема сообщений от пользователей, привести заявки в систематизированный вид, а также вести расширенную статистику решения заявок для анализа эффективности работы технической поддержки.

					ДП.09.02.07.ИР41.13.ПЗ	Лист
Изм.	Лист	№ докум.	Подпись	Дата		56



## 1 Основания для разработки

Разработка данной системы ведется на основании задания по подготовке выпускной квалификационной работы по специальности 09.02.07 «Информационные системы и программирование», которое рассмотрено на заседании цикловой комиссии вычислительных специальностей Кононовой О.А. и утверждено заместителем директора по учебно-производственной работе Лубниным С.Г. КОГПОБУ «Кировский авиационный техникум» на тему «Разработка веб-сайта для сбора заявок с проблемных участков»

Наименование темы разработки – «Веб-сайт для сбора заявок с проблемных участков».

					ДП.09.02.07.ИР41.13.ПЗ	Лист
Изм.	Лист	№ докум.	Подпись	Дата		57

## 2 Назначение разработки

### 2.1 Функциональное назначение

Веб-сайт предоставляет возможность работать с заявками от клиентов. Отвечать на них, закрывать, менять их статус. Есть возможность просмотра базы клиентов и статистики по решенным заявкам.

### 2.2 Эксплуатационное назначение

Программа должна эксплуатироваться в ОАО «Коммунэнерго» следующими пользователями: специалист технической поддержки, пользователь услугами компании, администратор.

					ДП.09.02.07.ИР41.13.ПЗ	Лист
Изм.	Лист	№ докум.	Подпись	Дата		58

### 3 Требование к программе или программному изделию

#### 3.1 Требования к функциональным характеристикам

Входные данные:

- данные о сотрудниках;
- сообщения клиентов;
- сообщения сотрудников;
- данные о статусе диалога.

Функции вебсайта:

- возможность отправить сообщение со стороны клиента и со стороны сотрудника технической поддержки;
- управление статусами заявок;
- разграничение прав доступа;
- создание графика, отражающего деятельность сотрудника технической поддержки, отображение написанных ранее сообщений;
- возможность выгрузить отчет в формате xls и pdf по статусам и датам заявок для всех сотрудников поддержки;
- поиск заявок и пользователей по фильтрам.

Выходные данные:

- график;
- таблицы в формате pdf и xls;
- отображение зарегистрированных пользователей и заявок;
- автоматическое назначение заявки на сотрудника поддержки после ответа на заявку.

#### 3.2 Требования к надежности

Так как в базе хранятся данные о клиентах, а также история диалогов, база данных подлежит резервному копированию.

Надежное и бесперебойное функционирование веб-сайта зависит от организационно-технических мероприятий, которые перечислены ниже:

- организацией бесперебойного питания сервера, на котором будет установлен веб-сайт;

					ДП.09.02.07.ИР41.13.ПЗ	Лист
Изм.	Лист	№ докум.	Подпись	Дата		59

- использованием лицензионного программного обеспечения;
- регулярным выполнением рекомендаций Министерства труда и социального развития РФ, изложенных в Постановлении от 23 июля
- регулярным выполнением требований ГОСТ 51188-98. Защита информации. Испытания программных средств на наличие компьютерных вирусов.

Время восстановления после отказа, вызванного сбоем электропитания технических средств, не фатальным сбоем операционной системы, не должно превышать 10 минут при условии соблюдения условий эксплуатации технических и программных средств.

Время восстановления после отказа, вызванного неисправностью технических средств, фатальным сбоем операционной системы, не должно превышать времени, требуемого на устранение неисправностей технических средств и переустановки программных средств.

### 3.3 Условия эксплуатации

Веб-сайт можно открыть на любом компьютере через браузер, так как все файлы расположены на удаленном сервере. База данных также хранится на сервере. Так как сайт находится на удаленном сервере должно быть обеспечено устойчивое интернет-соединение.

Предоставляется возможность открыть окно программы на весь экран, а также возможность закрыть, свернуть приложение или запустить любое стороннее программное обеспечение.

Специальных климатических условий эксплуатации, а также проведения каких-либо видов обслуживания не требуется.

При установке сайта на хостинг необходим системный администратор.

В процессе использования веб-сайта с ним взаимодействуют:

- клиенты ОАО «Коммунэнерго». Примером клиента может выступить человек, проживающий в многоквартирном доме, в котором электричество подключено ОАО «Коммунэнерго»;
- администратор, который отвечает за назначение зарегистрированного пользователя сотрудником технической поддержки;
- сотрудник технической поддержки, отвечающий на вопросы клиентов и работающий с заявками.

Администратор должен уметь работать в панели администратора Django.

					ДП.09.02.07.ИР41.13.ПЗ	Лист
						60
Изм.	Лист	№ докум.	Подпись	Дата		

Пользователи программы: администратор, специалист технической поддержки, клиенты должны обладать практическими навыками работы с графическим пользовательским интерфейсом браузера и операционной системы.

### 3.4 Требования к составу и параметрам технических средств

Состав технических средств:

Рекомендованные требования компьютера, с которого происходит доступ к сайту:

- процессор: 1,5 гГц или выше;
- оперативная память: 2 гб или выше;
- жесткий диск: 250 гб или выше;
- видеокарта с поддержкой opengl 2.0 или выше;
- операционная система: windows 7 или выше, macos x 10.7 или выше, linux с графической оболочкой.

Рекомендованные требования для сервера, на котором будут установлены веб-сайт и база данных:

- процессор: четырехъядерный процессор с тактовой частотой от 2,0 ГГц и выше;
- оперативная память: минимум 4 ГБ оперативной памяти;
- хранилище: жесткий диск или твердотельный накопитель (SSD) емкостью от 100 ГБ и более;
- сетевой интерфейс: Ethernet-адаптер с поддержкой скорости передачи данных от 1 Гбит/с и выше;
- операционная система: Linux Server 64bit;
- безопасность: сервер должен быть защищен от внешних атак (например, с помощью брандмауэра) и иметь механизмы резервного копирования данных для восстановления после сбоев.

### 3.5 Требования к информационной и программной совместимости

Для эксплуатации сайта необходим современный браузер, такой как Google Chrome, Yandex, Mozilla Firefox, Microsoft Edge и подобные. Также необходима операционная система с графическим интерфейсом. Примером таких систем могут выступить: Windows 7 и выше, Linux системы на Gnome, Mac OS и другие.

					ДП.09.02.07.ИР41.13.ПЗ	Лист
Изм.	Лист	№ докум.	Подпись	Дата		61

### 3.6 Специальные требования

Требования к сайту включают графический пользовательский интерфейс, который должен соответствовать рекомендациям и обеспечивать комфортное взаимодействие с пользователем.

Требования к графическому интерфейсу:

- интерфейс сайта должен быть выполнен в двух цветовых схемах: светлой и темной.
- цвета, которые можно использовать для интерфейса: #54ACFE, #4bc0c0, #ff6384, #3ba2eb, а также оттенки логотипа.

Необходимо обеспечить высокий уровень защиты данных и быстрый доступ к необходимой информации.

Веб-сайт обменивается с СУБД данными на сервере. На хостинге должен быть установлен Python в виртуальной среде `venv`, а также библиотека Django версии 4.2 и выше.

					ДП.09.02.07.ИР41.13.ПЗ	Лист
Изм.	Лист	№ докум.	Подпись	Дата		62

## 4 Требования к программной документации

Состав программной документации должен включать в себя:

- программный код разрабатываемого проекта;
- техническое задание в соответствии с ГОСТ 19.201-79;
- руководство оператора в соответствии с ГОСТ 19.505;
- функциональная структура системы;
- результаты оценки качества системы.

Документы должны быть выполнены в соответствии с ГОСТами к каждому виду документов (вышеперечисленные пункты).

					ДП.09.02.07.ИР41.13.ПЗ	Лист
Изм.	Лист	№ докум.	Подпись	Дата		63

## 5 Технико-экономические показатели

Создание сайта со сбором заявок с проблемных участков имеет множество плюсов. Во-первых, такой сайт позволит упростить процесс приёма заявок, что может значительно сэкономить время и силы сотрудников компании. Кроме того, создание сайта повысит уровень обслуживания клиентов, так как они смогут быстро и удобно оставлять заявки на решение проблемных вопросов, а также обеспечит возможность оперативного реагирования на поступающие заявки, что позволит компании обеспечивать высокий уровень качества услуг и удовлетворять потребности своих клиентов.

Кроме того, создание такого сайта может привести к экономии средств, которые ранее были затрачены на организацию и обслуживание телефонной линии для приёма заявок. Также, сайт сможет обрабатывать большее количество заявок в единицу времени, что повысит производительность работы компании.

					ДП.09.02.07.ИР41.13.ПЗ	Лист
Изм.	Лист	№ докум.	Подпись	Дата		64



## 6 Стадии и этапы разработки

Стадии разработки:

- разработка технического задания;
- разработка веб-сайта и документации к нему;
- тестирование веб-сайта;
- внедрение веб-сайта в эксплуатацию.

Этапы разработки:

- постановка цели;
- формирование и уточнение требований к техническому и программному обеспечению веб-сайта;
- определение функций веб-сайта;
- определение этапов и сроков разработки веб-сайта;
- выбор языка или языков программирования, способов хранения и передачи данных;
- согласование и утверждение технического задания.

На стадии разработки выполняются следующие этапы:

- разработка интерфейса веб-сайта;
- создание функциональной части веб-сайта;

На стадии тестирования выполняются следующие этапы:

- тестирование веб-сайта и определение недостатков;
- написание документации.

На стадии внедрения выполняют следующие этапы:

- подключение компонентов к базе данных;
- установка сайта на хостинг.

## 7 Порядок контроля и приемки

Перед началом приемки необходимо провести предварительную проверку, которая включает в себя проверку всех функций и характеристик веб-сайта, а также проверку соответствия веб-сайта техническому заданию.

На этапе тестирования проводятся тесты на соответствие системы требованиям, предъявляемым к веб-сайту. Проверяются все функции, а также безопасность и надежность. Тестирование проводится силами Исполнителя.

На этапе приемки проверяется соответствие системы всем требованиям и спецификациям, которые были определены в техническом задании.

После прохождения всех этапов контроля и приемки веб-сайт может быть сдан в эксплуатацию.

					ДП.09.02.07.ИР41.13.ПЗ	Лист
Изм.	Лист	№ докум.	Подпись	Дата		66

Приложение Г  
(обязательное)  
Программный код сайта

Листинг 1 – Формирование данных для отображения страницы «Заявки»  
(python)

```
from datetime import datetime, time
from time import sleep
from urllib.parse import unquote_plus
from django.forms.models import model_to_dict
from django.conf import settings
from django.contrib.auth.decorators import login_required
from django.contrib.auth.models import User, Group
from django.db.models import Q
from django.http import JsonResponse
from django.shortcuts import render, get_object_or_404, redirect
from django.urls import reverse
import imghdr
import json
from .forms import MessageForm, ConversationForm
from .models import Conversation, Message, Attachment
from django.db.models import Max, Subquery, OuterRef

@login_required
def conversations_list(request):
    if request.method == 'POST':
        form = ConversationForm(request.POST)
        if form.is_valid():
            new_conversation = form.save(commit=False)
            new_conversation.user1 = request.user
            new_conversation.status = 'Активная'
            new_conversation.save()
```

```

        return JsonResponse({'url': reverse('conversation_detail',
args=[new_conversation.id])})
    else:
        return JsonResponse({'error': form.errors})
    else:
        form = ConversationForm()

user = request.user
if user.is_authenticated and user.groups.filter(name='Обычные пользователи').exists():
    normal_users = User.objects.filter(
        groups__name='Обычные пользователи')
    conversations = Conversation.objects.filter(
        Q(user1__in=normal_users) | Q(user2__in=normal_users))
else:
    conversations = Conversation.objects.all()

number = request.GET.get('number')
date = request.GET.get('date')
status = request.GET.get('status')
sender = request.GET.get('sender')
recipient = request.GET.get('recipient')
context = {}

if number:
    try:
        number = int(number)
        conversations = conversations.filter(id__icontains=number)
        context['number'] = number
    except:
        context['number'] = 'Введите число'

if date:
    date1 = date

```

```

date = datetime.strptime(date, '%Y-%m-%d').date()
start_date = datetime.combine(date, time.min)
end_date = datetime.combine(date, time.max)
conversations = conversations.filter(
    created_at__gte=start_date, created_at__lt=end_date)
context['date'] = date1

```

if status:

```

conversations = conversations.filter(status=status)
context['status'] = status

```

if sender:

```

sender = unquote_plus(sender)
sender_parts = sender.split()
if len(sender_parts) >= 2:
    sender_filter = Q(user1__first_name__icontains=sender_parts[0]) | Q(
        user1__last_name__icontains=sender_parts[1])
else:
    sender_filter = Q(user1__first_name__icontains=sender) | Q(
        user1__last_name__icontains=sender)
conversations = conversations.filter(sender_filter)
context['sender'] = sender

```

if recipient:

```

recipient = unquote_plus(recipient)
recipient_parts = recipient.split()
if len(recipient_parts) >= 2:
    recipient_filter = Q(user2__first_name__icontains=recipient_parts[0]) | Q(
        user2__last_name__icontains=recipient_parts[1])
else:
    recipient_filter = Q(user2__first_name__icontains=recipient) | Q(
        user2__last_name__icontains=recipient)
conversations = conversations.filter(recipient_filter)
context['recipient'] = recipient

```

```

print(recipient)

conversations = conversations.annotate(
    latest_message_time=Subquery(
        Message.objects.filter(conversation=OuterRef('pk')).values('created_at').order_by('-
created_at')[:1]
    )).order_by('-latest_message_time')

context.update({
    'conversations': conversations
})

print(context)

return render(request, 'messages_home/conversations_list.html', context=context)

```

@login\_required

```

def conversation_detail(request, conversation_id):
    conversation = get_object_or_404(Conversation.objects.filter(Q(id=conversation_id)))
    messages = conversation.messages.all()
    attachments = Attachment.objects.filter(message__conversation=conversation)
    support_group = Group.objects.get(name='Поддержка')
    support_users = User.objects.filter(groups__name=support_group.name)

    if request.method == 'POST':
        form = MessageForm(request.POST, request.FILES)
        if form.is_valid():
            message = form.save(commit=False)
            message.conversation = conversation
            message.sender = request.user
            attachment = request.FILES.get('message_attachment')
            if attachment is not None:
                attachment = Attachment.objects.create(

```

```

        file=attachment,
        uploaded_by=request.user
    )
message.attachment = attachment

if request.user.groups.filter(name='Поддержка').exists():
    conversation.user2 = message.sender
conversation.save()
message.save()

created_at = message.created_at
formatted_date = created_at.strftime('%d %B %Y г. %H:%M')
formatted_date = formatted_date.replace(
formatted_date.split()[1],
{
    "January": "января",
    "February": "февраля",
    "March": "марта",
    "April": "апреля",
    "May": "мая",
    "June": "июня",
    "July": "июля",
    "August": "августа",
    "September": "сентября",
    "October": "октября",
    "November": "ноября",
    "December": "декабря",
}[formatted_date.split()[1]],)

if message.attachment is not None:
    message_data = {
        'attachment': message.attachment.file.url,
        'text': message.text,

```

```

        'sender': message.sender.username,
        'sender_first_name': message.sender.first_name,
        'sender_last_name': message.sender.last_name,
        'created_at': formatted_date,
        'user': request.user.username,
    }
    print(message.sender.username)
else:
    message_data = {
        'text': message.text,
        'sender': message.sender.username,
        'sender_first_name': message.sender.first_name,
        'sender_last_name': message.sender.last_name,
        'created_at': formatted_date,
        'user': request.user.username,
    }

    return JsonResponse({'message': message_data})
else:
    form = MessageForm()

context = {
    'conversation': conversation,
    'messages': messages,
    'form': form,
    'user': request.user,
    'support_users': support_users,
}

return render(request, 'messages_home/conversation_detail.html', context)

@login_required
def message_create(request):
    if request.method == 'POST':

```



```

form = MessageForm(request.POST, request.FILES)
if form.is_valid():
    message = form.save(commit=False)
    message.sender = request.user
    message.save()
    data = {'is_valid': True, 'name': message.attachment.name,
            'url': message.attachment.url}
else:
    data = {'is_valid': False}
else:
    data = {'is_valid': False}
return JsonResponse(data)

```

```

@login_required
def conversation_ajax(request, pk):
    if request.META.get('HTTP_X_REQUESTED_WITH') == 'XMLHttpRequest' and
request.method == 'GET':
        conversation_id = pk
        conversation = Conversation.objects.get(id=conversation_id)
        messages = Message.objects.filter(
            conversation=conversation).order_by('-created_at')[:10]
        message_list = []
        for message in messages:
            message_list.append(
                {'text': message.text, 'created_at': message.created_at, 'sender':
message.sender.username})
        return JsonResponse({'messages': message_list})
    else:
        return JsonResponse({'error': 'Invalid request'})

```

```

@login_required
def conversation_new(request):

```

```

if request.method == 'POST':
    form = ConversationForm(request.POST)
    if form.is_valid():
        conversation = form.save(commit=False)
        conversation.user1 = request.user # Устанавливаем user1
        conversation.save()
        return redirect('conversation_detail', conversation_id=conversation.id)
    else:
        form = ConversationForm()
return render(request, 'messages_home/new_conversation.html', {'form': form})

```

@login\_required

```

def update_conversation_user(request, conversation_id):
    body_unicode = request.body.decode('utf-8')
    data = json.loads(body_unicode)
    user_id = data.get("user_id")
    conversation = Conversation.objects.get(pk=conversation_id)
    try:
        user = User.objects.get(first_name=user_id.split(' ')[0], last_name=user_id.split(' ')[1])
    except:
        user = None
    conversation.user2 = user
    conversation.save()
    return JsonResponse({"success": True})

```

@login\_required

```

def update_conversation_status(request, conversation_id):
    body_unicode = request.body.decode('utf-8')
    data = json.loads(body_unicode)
    status_id = data.get("selected_status")
    conversation = Conversation.objects.get(pk=conversation_id)
    conversation.status = str(status_id)
    conversation.save()
    return JsonResponse({"success": True})

```

## Листинг 2 - Отрисовка графика с помощью chart.js

```
<script src="https://cdn.jsdelivr.net/npm/chart.js"></script>
<script>
    document.addEventListener("DOMContentLoaded", function () {
        var dataString = '{{ data|safe }}'
        var data = JSON.parse(dataString);

        var labels = [];
        var activeData = [];
        var closedData = [];
        var snoozedData = [];

        var uniqueLabels = Array.from(new Set(data.labels));
        uniqueLabels.sort(); // Сортировка дат по возрастанию

        for (var i = 0; i < uniqueLabels.length; i++) {
            var label = uniqueLabels[i];
            var activeCount = data.datasets[0].data.filter(function (date) {
                return date === label;
            }).length;
            var closedCount = data.datasets[1].data.filter(function (date) {
                return date === label;
            }).length;
            var snoozedCount = data.datasets[2].data.filter(function (date) {
                return date === label;
            }).length;

            labels.push(label);
            activeData.push(activeCount);
            closedData.push(closedCount);
            snoozedData.push(snoozedCount);
        }
    });
</script>
```

```

var chartData = {
  labels: labels,
  datasets: [
    {
      label: data.datasets[0].label,
      data: activeData,
      backgroundColor: "rgba(75, 192, 192, 0.2)",
      borderColor: "rgba(75, 192, 192, 1)",
      borderWidth: 1,
      tension: 0.4
    },
    {
      label: data.datasets[1].label,
      data: closedData,
      backgroundColor: "rgba(255, 99, 132, 0.2)",
      borderColor: "rgba(255, 99, 132, 1)",
      borderWidth: 1,
      tension: 0.4
    },
    {
      label: data.datasets[2].label,
      data: snoozedData,
      backgroundColor: "rgba(54, 162, 235, 0.2)",
      borderColor: "rgba(54, 162, 235, 1)",
      borderWidth: 1,
      tension: 0.4
    }
  ]
};

```

```

var options = {

```

```

responsive: true,
scales: {
  x: {
    title: {
      display: true,
      text: 'Дата',
      color: '#fff',
    },
    ticks: {
      color: '#fff',
    }
  },
  y: {
    title: {
      display: true,
      text: 'Количество заявок',
      color: '#fff',
    },
    ticks: {
      stepSize: 1,
      color: '#fff',
    }
  }
},
plugins: {
  title: {
    display: true,
    text: 'График заявок по датам',
    color: '#fff',
  },
  legend: {
    labels: {
      color: '#fff',

```

```

    }

    }
    }
};

var ctx = document.getElementById("chart").getContext("2d");
new Chart(ctx, {
    type: "line",
    data: chartData,
    options: options
});
});
</script>

```

					ДП.09.02.07.ИР41.13.ПЗ	Лист
						78
Изм.	Лист	№ докум.	Подпись	Дата		

Приложение Д  
(обязательное)  
Библиография

- 1 ГОСТ 2.105-95. Межгосударственный стандарт. Единая система конструкторской документации. Общие требования к текстовым документам (введен Постановлением Госстандарта от 08.08.1995 N 426) (ред. От 22.06.2006)
- 2 ГОСТ 2.316-2008. Единая система конструкторской документации (ЕСКД). Правила нанесения подписей, технических требований и таблиц на графических документах.
- 3 Информация о компании ОАО «Коммунэнерго» [Электронный ресурс] – Режим доступа: [https://xn--cladoiagaegs3a5j.xn--p1ai/about\\_company](https://xn--cladoiagaegs3a5j.xn--p1ai/about_company). – (Дата обращения 27.04.2023).
- 4 Определение программного обеспечения [Электронный ресурс]. – Режим доступа: <https://dic.academic.ru/dic.nsf/enc1p/39050>. – (Дата обращения 2.05.2023).
- 5 Определение бизнес-процесса [Электронный ресурс]. – Режим доступа: <https://www.bptrends.com/harmon-on-bpm-whats-next-for-business-process>. – (Дата обращения: 5.05.2023).
- 6 Определение автоматизации бизнес-процессов [Электронный ресурс]. – Режим доступа: <https://www.techtarget.com/searchcio/definition/business-process-automation>. – (Дата обращения: 5.05.2023).
- 7 Дронов В.П. Django 2.1. Практика создания веб-сайтов на Python. - М.: Django, 2019. - 480 с.
- 8 Percival H. Test-Driven Development with Python: Obey the Testing Goat: Using Django, Selenium, and JavaScript 2nd Edition, Kindle Edition // Book. - 2017. - 614 с.
- 9 Brenda J. Web API Design: Crafting Interfaces that Developers Love // Book. - 2018. - 296 с.
- 10 Дакетт Дж. HTML и CSS. Разработка и дизайн веб-сайтов. - М.: издательство "ДМК Пресс", 2019. - 640 с.
- 11 Васвани В. SQLite: использование и администрирование. - М.: издательство "Питер", 2014. - 944 с.
- 12 Доусон М. Программирование на Python - М.: издательский дом "Вильямс", 2020. - 800с.

					ДП.09.02.07.ИР41.13.ПЗ	Лист
Изм.	Лист	№ докум.	Подпись	Дата		79