

Министерство науки и высшего образования Российской Федерации
Федеральное государственное бюджетное образовательное учреждение
высшего образования
АМУРСКИЙ ГОСУДАРСТВЕННЫЙ УНИВЕРСИТЕТ
(ФГБОУ ВО «АмГУ»)

Институт компьютерных и инженерных наук
Кафедра информационных и управляющих систем
Направление подготовки 09.03.01 – Информатика и вычислительная техника

ДОПУСТИТЬ К ЗАЩИТЕ

Зав. кафедрой

_____ А.В. Бушманов

« _____ » _____ 2024 г.

БАКАЛАВРСКАЯ РАБОТА

на тему: Разработка информационной системы «Электронная приёмная» для администрации села Константиновка

Исполнитель
студент группы 0103-об

(подпись, дата)

М.А. Смирнов

Руководитель
доцент, канд. техн. наук

(подпись, дата)

С.Г. Самохвалова

Консультант:
по безопасности и экологичности
доцент, канд. техн. наук

(подпись, дата)

А.Б. Булгаков

Нормоконтроль
инженер кафедры

(подпись, дата)

В.Н. Адаменко

Благовещенск 2024

Министерство науки и высшего образования Российской Федерации
Федеральное государственное бюджетное образовательное учреждение
высшего образования
АМУРСКИЙ ГОСУДАРСТВЕННЫЙ УНИВЕРСИТЕТ
(ФГБОУ ВО «АмГУ»)

Институт компьютерных и инженерных наук

Кафедра информационных и управляющих систем

УТВЕРЖДАЮ

Зав. кафедрой

_____ А.В. Бушманов

« _____ » _____ 2023 г.

ЗАДАНИЕ

К выпускной квалификационной работе студента Смирнов М.А.

1. Тема выпускной квалификационной работы: Разработка информационной системы «Электронная приёмная» для администрации села Константиновка
(утверждено приказом от 03.04.2024 № 890-уч)

2. Срок сдачи студентом законченной работы (проекта) _____

3. Содержание выпускной квалификационной работы: анализ предметной области; выбор среды разработки; разработка информационной системы; тестирование программного продукта; безопасность и экологичность.

4. Перечень материалов приложения: техническое задание

5. Дата выдачи задания 02.10.2023

Руководитель выпускной квалификационной работы: _____

Самохвалова С.Г. доцент, канд. техн. наук
(фамилия, имя, отчество, должность, уч.степень, уч.звание)

Задание принял к исполнению (02.10.23) _____
(подпись студента)

РЕФЕРАТ

Бакалаврская работа содержит 103 с., 76 рисунков, 10 таблиц, 20 источников.

ВЕРСТКА, РАЗРАБОТКА САЙТА, ВЕБ-РАЗРАБОТКА, PHP, CSS, HTML, JAVASCRIPT

Объектом исследования является процесс создания сайта администрации села Константиновка.

Целью работы является разработка сайта администрации с необходимым функционалом для работников.

Процесс выполнения работы разделен на несколько этапов:

Первый этап – анализ деятельности администрации села Константиновка;

Второй этап – анализ существующих решений;

Третий этап – выбор среды разработки;

Четвертый этап – проектирование базы данных, верстка сайта, написание серверной части;

Пятый этап – тестирование.

Результатом выполнения данной работы является сайт администрации села Константиновка.

СОДЕРЖАНИЕ

Введение	7
1 Анализ предметной области	10
1.1 Анализ деятельности Администрации села Константиновка	10
1.1.1 Основная информация об учреждении	10
1.1.2 Задачи и функции учреждения	11
1.1.3 Анализ организационной структуры учреждения	17
1.2 Структура документооборота	21
1.2.1 Внешний документооборот	21
1.2.2 Внутренний документооборот	23
1.3 Пример аналогичной информационной системы	26
2 Разработка информационной системы	31
2.1 Среда разработки	31
2.2 Проектирование баз данных	34
2.2.1 Инфологическое проектирование	35
2.2.2 Логическое проектирование	41
2.2.3 Физическое проектирование	48
2.4 Разработка сайта	51
2.5 Проектирование интерфейса	68
2.6 Тестирование программного обеспечения	72
3 Безопасность и экологичность	80
3.1 Безопасность	80
3.1.1 Требования к помещению для работы с ПЭВМ	81
3.1.2 Требования к освещению на рабочих местах с ПЭВМ	83
3.1.3 Требования к организации рабочих мест с ПЭВМ	84
3.1.4 Требования к микроклимату рабочего места	87
3.2 Экологичность	88
3.3 Чрезвычайные ситуации	91

Заключение	95
Библиографический список	96
Приложение А	99

НОРМАТИВНЫЕ ССЫЛКИ

В настоящей бакалаврской работе использованы ссылки на следующие стандарты и нормативные документы:

ГОСТ 19.004-80. ЕСПД. Термины и определения.

ГОСТ ISO 9001-2011 Системы менеджмента качества. Требования.

ГОСТ 2.052 2015 Единая система конструкторской документации.

Электронная модель изделия. Общие положения

ГОСТ 24105-80 Изделия из пластмасс. Термины и определения дефектов

ГОСТ 12.0.003-2015 Система стандартов безопасности труда (ССБТ).
Опасные и вредные производственные факторы. Классификация.

ВВЕДЕНИЕ

Современные технологии могут значительно улучшить деловые процессы в органах местного самоуправления при минимальных затратах, создавая механизмы для быстрого и эффективного взаимодействия между населением и властью. Это позволяет не только оптимизировать работу администраций, но и повысить прозрачность и подотчетность их деятельности.

Одним из наиболее действенных инструментов для достижения этих целей является внедрение электронной общественной приемной. Эта платформа, интегрированная с веб-службой и системой документооборота администрации, предоставляет гражданам удобный и доступный способ взаимодействия с органами власти. Граждане могут оперативно направлять свои обращения, запросы и предложения, отслеживать их рассмотрение и получать обратную связь в реальном времени.

Такая система способствует вовлечению населения в процесс управления и принятия решений, что повышает уровень доверия к местным органам власти. Администрации получают возможность более оперативно реагировать на нужды и проблемы граждан, улучшать качество предоставляемых услуг и быстрее адаптироваться к изменениям и новым вызовам.

Кроме того, использование современных технологий в органах местного самоуправления способствует экономии ресурсов и снижению административных затрат. Автоматизация и цифровизация процессов позволяют сократить бумажный документооборот, уменьшить количество рутинных задач и освободить время для более значимых и стратегически важных проектов.

Таким образом, внедрение современных технологий и создание механизмов оперативного взаимодействия между населением и властью является ключевым фактором для эффективного функционирования органов местного самоуправления, повышения качества жизни граждан и укрепления демократии на местном уровне.

Ежегодно граждане по всей стране тратят около трех миллиардов человеко-часов на взаимодействие с представителями власти, что составляет примерно 2 % их времени. Внедрение информационных технологий в этот процесс может существенно сократить временные затраты.

Электронная приемная представляет собой, с одной стороны, виртуальное пространство, которое включает несколько страниц на сайте организации, запрограммированных для выполнения функций по приему заявлений граждан и размещению справочной информации.

С другой стороны, это программное обеспечение, которое позволяет гражданам:

- получать доступ к сетевым ресурсам (как автономному сайту администрации, так и подключенному к Интернету);
- без необходимости стоять в длинных очередях своевременно получать необходимую информацию о работе специалистов;
- автоматизировать документооборот, включая прием писем от граждан и их передачу в соответствующие инстанции;
- задавать вопросы, оставлять предложения и узнавать последние новости.

Таким образом, электронную приемную можно охарактеризовать как систему автоматизации процесса предоставления государственных и муниципальных услуг.

Объект исследования – Муниципальное учреждение Администрация села Константиновка.

Целью бакалаврской работы является создание электронной приемной на основе web-технологий для предоставления пользователям возможности интерактивного взаимодействия с должностными лицами муниципалитета, свободного обсуждения проблем общественной жизни и непосредственного участия жителей в их решении.

Для реализации поставленной цели необходимо решить следующие за-

дачи:

- исследовать предметную область;
- проанализировать основные показатели учреждения;
- проанализировать внутренний и внешний документооборот;
- произвести выбор среды разработки и программных продуктов;
- разработать проект информационной подсистемы.

1 АНАЛИЗ ПРЕДМЕТНОЙ ОБЛАСТИ

1.1 Анализ деятельности Администрации села Константиновка

1.1.1 Основная информация об учреждении

Местное самоуправление в селе Константиновка Константиновского района Амурской области представляет собой форму реализации гражданами своей власти на местном уровне. Благодаря этому население может самостоятельно и ответственно решать вопросы местного значения в соответствии с Конституцией РФ, федеральными законами и законодательством Амурской области. Это происходит напрямую или через органы местного самоуправления с учётом интересов жителей и исторических особенностей региона.

Муниципальное учреждение Администрация села Константиновка – исполнительно-распорядительный орган местного самоуправления поселения и является юридическим лицом.

Адрес месторасположения Администрации села Константиновка: 676980, Амурская область, Константиновский район, с. Константиновка, ул. Ленина, д. 85, кв. 41.

Территория муниципального образования Константиновского района Амурской области находится на востоке России. Константиновский район расположен в центральной части Амурской области и включает в себя несколько населенных пунктов, таких как село Константиновка и другие. Площадь и границы района можно уточнить на официальном сайте администрации района или в специализированных источниках информации о регионах России.

Финансовое обеспечение деятельности Администрации села Константиновка и ее органов осуществляется за счет средств бюджета села Константиновка.

Правовую базу местного самоуправления образуют: Конституция РФ, федеральные конституционные законы, федеральные законы и вытекающие

из них другие нормативно-правовые акты Российской Федерации, законы и иные нормативные акты Амурской области, решения, принятые на местных референдумах, а также другие муниципальные правовые акты.

1.1.2 Задачи и функции учреждения

Основополагающими задачами администрации села Константиновка являются организация эффективного местного самоуправления и реализация полномочий по комплексному решению вопросов, имеющих первостепенное значение для жителей данного муниципального образования. В полномочия органов местного самоуправления входят:

- разработка проекта бюджета поселения, его всестороннее рассмотрение, утверждение и исполнение, осуществление контроля за расходованием бюджетных средств и подготовка отчета об исполнении бюджета для обеспечения прозрачности и подотчетности; установление, изменение и отмена местных налогов и сборов поселения;

- администрация сельского поселения владеет, пользуется и распоряжается имуществом, находящимся в муниципальной собственности. Она эффективно управляет этими ресурсами, чтобы обеспечить интересы жителей.

- в рамках поселения администрация отвечает за электро-, тепло-, водоснабжение населения, водоотведение и снабжение жителей топливом в соответствии с законодательством;

- администрация заботится о содержании, ремонте и строительстве автомобильных дорог местного значения, обустройстве парковочных мест и обеспечении безопасности дорожного движения. Она также контролирует состояние дорог и решает другие вопросы, связанные с управлением дорожной инфраструктурой;

- одна из важных функций администрации – предоставление жилья из муниципального фонда малоимущим гражданам, которые нуждаются в улучшении жилищных условий. Администрация строит и содержит муници-

пальное жильё и создаёт условия для развития жилищного строительства, а также осуществляет муниципальный жилищный контроль;

– администрация села также отвечает за предоставление транспортных услуг населению и организацию эффективного транспортного обслуживания в пределах поселения. Она обеспечивает доступность различных видов общественного транспорта для жителей;

– местная власть уделяет большое внимание укреплению межнационального и межрелигиозного согласия, сохранению и развитию культурных традиций и языков коренных народов. Она также содействует социальной и культурной адаптации мигрантов и предотвращению межнациональных и межэтнических конфликтов. Администрация стремится создать атмосферу взаимного уважения и толерантности в поселении;

– администрация участвует в борьбе с терроризмом и экстремизмом, минимизирует последствия этих явлений и ликвидирует их проявления в границах поселения. Она также принимает превентивные меры по предупреждению и ликвидации чрезвычайных ситуаций на своей территории;

– одной из важных задач администрации является обеспечение первичных мер пожарной безопасности в населённых пунктах села. Она создает необходимые условия для оперативного реагирования и эффективного предотвращения пожарных рисков;

– кроме того, администрация создает условия для предоставления жителям качественных услуг связи, общественного питания, торговли и бытового обслуживания. Она содействует развитию соответствующей инфраструктуры и поддерживает местных предпринимателей, работающих в этих сферах;

– местные власти стремятся предоставить жителям доступ к услугам связи, общественного питания, торговли и бытовых услуг. Для этого они оказывают поддержку открытию и развитию соответствующих предприятий, а также создают необходимую инфраструктуру;

– одна из основных задач администрации – организация качественного библиотечного обслуживания населения. Она отвечает за пополнение и сохранение библиотечных фондов, обновление книжных коллекций, а также поддержание библиотек в надлежащем состоянии как центров просвещения и культурного досуга жителей;

– кроме того, местная власть стремится создать условия для организации разнообразного досуга и обеспечения жителей услугами учреждений культуры. Она занимается сохранением, использованием и популяризацией объектов культурного наследия, находящихся в собственности поселения. Администрация также осуществляет охрану объектов культурного наследия местного значения, расположенных на территории;

– местная власть также поддерживает развитие местного традиционного народного художественного творчества. Она участвует в сохранении, возрождении и развитии народных художественных промыслов в поселении, оказывая поддержку мастерам и ремесленникам, а также организуя фестивали и выставки народного искусства;

– администрация населённого пункта создаёт условия для развития физической культуры, школьного спорта и массового спорта на своей территории. Она организует официальные физкультурно-оздоровительные и спортивные мероприятия для местных жителей, такие как соревнования, спартакиады и праздники. Это помогает привлекать население к занятиям физкультурой и спортом;

– местные власти обеспечивают условия для массового отдыха жителей. Они обустривают места для отдыха, например парки, скверы и пляжи. Администрация также обеспечивает свободный доступ граждан к общим водным объектам и их береговым полосам, обустривая и поддерживая в хорошем состоянии пляжи и другие зоны отдыха возле водоёмов;

– администрация занимается созданием и поддержкой архивных фондов, которые сохраняют историю развития и жизни территории. Она собира-

ет, хранит, учитывает и использует документы, отражающие социально-экономическое, общественно-политическое и культурное развитие населённого пункта;

– местные власти участвуют в организации процесса сбора и перевозки твёрдых коммунальных отходов, что обеспечивает экологическую безопасность и чистоту территории;

– кроме того, администрация устанавливает стандарты благоустройства территории. Они включают требования к содержанию зданий, сооружений и земельных участков, на которых они находятся, к внешнему виду фасадов и ограждений. Стандарты также определяют перечень работ по благоустройству, их периодичность и порядок участия владельцев зданий и сооружений в благоустройстве прилегающих территорий. Это помогает поддерживать единый эстетический облик населённых пунктов;

– непосредственно организация благоустройства территории поселения является еще одной важной функцией местной администрации. Это включает освещение улиц, озеленение, установку указателей с названиями улиц и номерами домов, размещение и обслуживание малых архитектурных форм. Кроме того, в ее ведение входит использование, защита и восстановление городских лесов и особо охраняемых природных территорий, расположенных в границах населенных пунктов, утверждение генеральных планов поселения, правил землепользования и застройки, утверждение подготовленной на основе генеральных планов поселения документации по планировке территории, выдача разрешений на строительство (за исключением случаев, предусмотренных Градостроительным кодексом Российской Федерации, иным федеральными законами), разрешений на ввод объектов в эксплуатацию при осуществлении строительства, реконструкции объектов капитального строительства, расположенных на территории поселения, утверждение местных нормативов градостроительного проектирования поселений, резервирование земель и изъятие земельных участков в границах поселения для муниципаль-

ных нужд, осуществление муниципального земельного контроля в границах поселения, осуществление в случаях, предусмотренных Градостроительным кодексом Российской Федерации, осмотров зданий, сооружений и выдача рекомендаций об устранении выявленных в ходе таких осмотров нарушений;

– присвоение адресов объектам адресации, изменение, аннулирование адресов, присвоение наименований элементам улично-дорожной сети (за исключением автомобильных дорог федерального значения, автомобильных дорог регионального или межмуниципального значения, местного значения муниципального района), наименований элементам планировочной структуры в границах поселения, изменение, аннулирование таких наименований, размещение информации в государственном адресном реестре;

– организация похоронных услуг и уход за местами захоронений;

– организация и осуществление мероприятий по территориальной обороне и гражданской обороне, защите населения и территории поселения от чрезвычайных ситуаций природного и техногенного характера;

– создание, содержание и организация деятельности аварийно-спасательных служб и (или) аварийно-спасательных формирований на территории поселения;

– администрация посёлка реализует меры по обеспечению безопасности людей на водных объектах и охране их здоровья и жизни. В частности, это может включать установку спасательных станций, обустройство пляжей и зон отдыха на воде, организацию дежурства спасателей в пиковый сезон купания, а также распространение информации о правилах поведения и безопасности на водоёмах;

– местные власти создают, развивают и поддерживают охрану лечебно-оздоровительных зон и местных курортов, находящихся на территории посёлка. Это может включать благоустройство и поддержание в надлежащем состоянии источников минеральной воды, грязелечебниц, пляжей и других

инфраструктурных объектов. Администрация обеспечивает надлежащие условия для отдыха и оздоровления жителей;

– кроме того, администрация осуществляет муниципальный контроль над использованием и охраной особо охраняемых природных территорий местного значения. Она следит за соблюдением установленных режимов и правил использования данных территорий;

– администрация также способствует развитию сельского хозяйства, предоставляя земельные участки для фермерских хозяйств, консультируя и финансово поддерживая местных производителей сельскохозяйственной продукции. Это позволяет стимулировать рост сельхозпроизводства в поселении;

– также она создаёт условия для развития малого и среднего бизнеса, предлагая льготы, преференции и информационную поддержку предпринимателям. Это способствует росту деловой активности и созданию новых рабочих мест;

– администрация организует и проводит различные мероприятия для детей и молодёжи, такие как спортивные соревнования, культурные и досуговые программы, профориентационные мероприятия, лагеря и секции. Эта работа направлена на всестороннее развитие подрастающего поколения;

– администрация предоставляет помещение для работы на обслуживаемом административном участке поселения сотруднику, замещающему должность участкового уполномоченного полиции. Это способствует повышению эффективности участковой службы и обеспечению правопорядка на территории;

– оказание поддержки социально ориентированным некоммерческим организациям в пределах полномочий, установленных статьями 31.1 и 31.3 Федерального закона от 12 января 1996 года №7-ФЗ «О некоммерческих организациях»;

– осуществление муниципального контроля на территории особой экономической зоны;

– осуществление мер по противодействию коррупции в границах поселения.

Таким образом, Администрация села Константиновка выполняет большой объем функций. Создание электронной приемной не только сократит затраты учреждения на обработку поступающей от потребителей информации, но и существенно улучшит качество сервиса для граждан.

1.1.3 Анализ организационной структуры учреждения

На январь 2024 года штат учреждения составляют 34 сотрудника.

Администрацией села Константиновка руководит глава села Константиновка на принципах единоначалия, таким образом организационно-управленческая структура учреждения является линейной.

Организационно-управленческая структура учреждения изображена на рисунке 1.

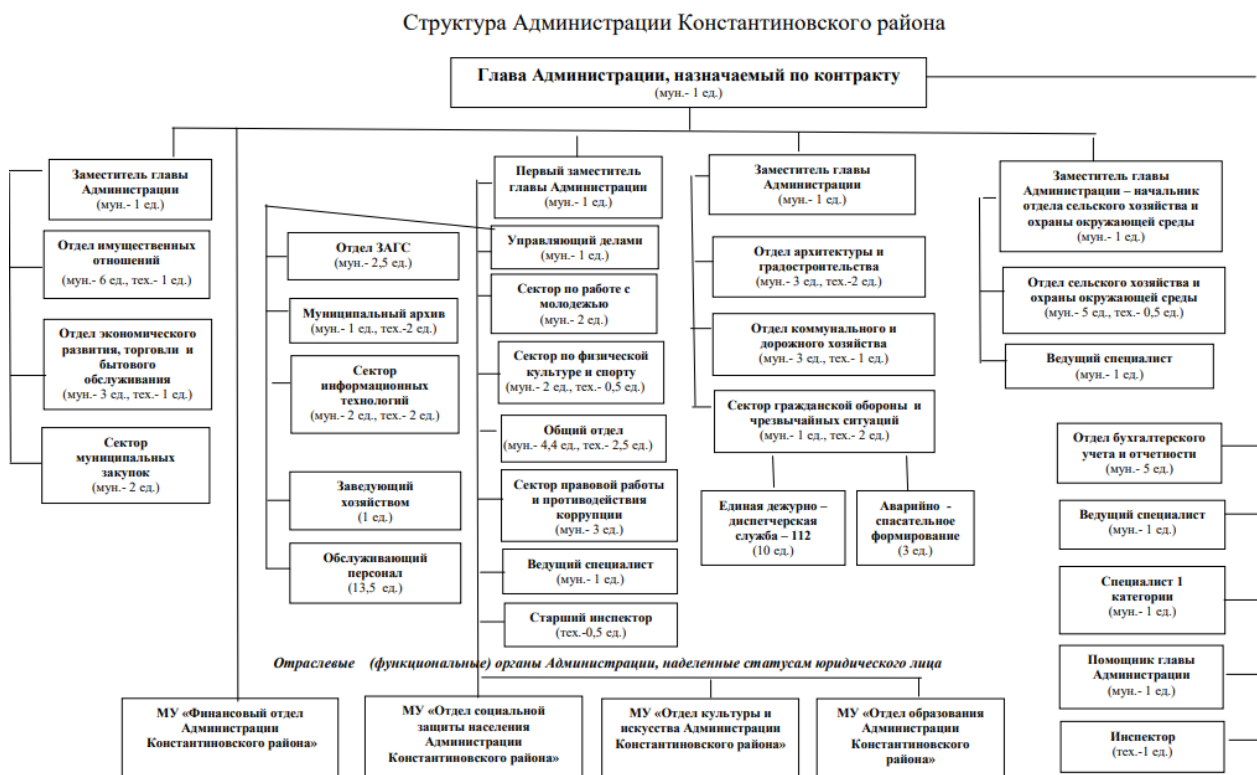


Рисунок 1 – Организационно-управленческая структура Администрации Константиновского района

Глава села Константиновка обладает широким спектром полномочий, включая руководство деятельностью Администрации в соответствии с уставом, назначение и освобождение от должности муниципальных служащих, принятие мер по поощрению и дисциплинарной ответственности, защиту интересов поселения в различных инстанциях, подписание договоров и соглашений, представление проектов бюджета и других нормативных актов перед представительным органом, организацию социально-экономического развития, решение вопросов аварийной ситуации и представление интересов села в различных сферах.

Глава села Константиновка представляет Константиновскому поселковому Совету народных депутатов и гражданам ежегодные отчеты о результатах деятельности администрации села Константиновка, в том числе о решении вопросов, поставленных Константиновским поселковым Советом народных депутатов.

Главе села Константиновка подчиняются муниципальные служащие, и все остальные группы.

К основным обязанностям муниципальных служащих относятся:

- соблюдение законности в деятельности администрации и ведение юридической работы; осуществление юридической защиты интересов администрации; организация договорной, претензионной и исковой работы;

- анализ социально-экономического развития села; разработка целевых программ, мониторинг эффективности реализации программ; составление реестра бюджетных обязательств по состоянию на 01 июля текущего года и на 15 февраля следующего за отчетным годом; планирование бюджета села Константиновка;

- осуществление подбора и размещения персонала в администрации включает в себя организацию конкурсов на заполнение открытых вакансий в муниципальной службе, а также управление процессом аттестации сотрудников администрации. Кроме того, в обязанности входит регистрация входящей

корреспонденции от государственных и местных органов власти, организаций и граждан, а также учет и обработка обращений, заявлений и жалоб жителей с последующим их рассмотрением и ответом. Документы, поступающие на регистрацию, ведутся в специальной картотеке, отслеживается их выполнение, и выдаются необходимые справки по ним. Безопасность и сохранность всех документов важна и требует особого внимания;

– предоставление гражданам необходимых справок в соответствии с законодательством, размещение информации на веб-сайте администрации, а также предоставление характеристик жителей по запросам соответствующих органов и ведение учета граждан, нуждающихся в жилье, – все это входит в задачи администрации села Константиновка.

К основным должностным обязанностям группы относятся:

– организация бухгалтерского учета, который включает в себя отслеживание исполнения сметы доходов и расходов администрации, является неотъемлемой частью ее деятельности. Контроль за эффективным и законным использованием бюджетных средств, строго соблюдая их целевое назначение, также важен. Важным аспектом является правильное начисление и своевременное перечисление налогов, платежей и страховых взносов. Кроме того, в обязанности администрации входит своевременное и качественное представление разнообразной отчетной документации, включая бухгалтерскую, статистическую и налоговую отчетность;

– разработка законов и правил, планирование и проведение мероприятий по гражданской обороне и предотвращению пожаров представляют собой важный аспект обеспечения безопасности и защиты населения и территорий. Эти действия включают в себя не только разработку законодательства и нормативных актов, но и организацию системы планирования и реагирования на чрезвычайные ситуации. Кроме того, значительное внимание уделяется созданию, хранению и эффективному использованию резервов финансовых и материальных ресурсов, необходимых для оперативного реагирования

на чрезвычайные ситуации. Обучение населения различным методам защиты и безопасности, а также подготовка молодежи к действиям в экстремальных ситуациях, играют ключевую роль в повышении общественной готовности и реагирования на угрозы. Также, координация деятельности аварийно-спасательных служб и организаций, действующих на территории муниципального образования, обеспечивает эффективное взаимодействие и согласованность действий в чрезвычайных ситуациях.

Группа по земле и имуществу отвечает за ряд ключевых обязанностей. В их числе – защита прав и интересов граждан в сфере земельных вопросов, взаимодействие с налоговыми органами для предоставления им точной информации о земельных и имущественных вопросах, находящихся в собственности граждан или юридических лиц, с целью формирования налогооблагаемой базы по муниципальному образованию. Они также занимаются разработкой муниципальной политики в области земельных и имущественных вопросов, ведением списков бесхозного и выморочного имущества, а также аварийного жилья и муниципального жилого фонда. Группа осуществляет муниципальный земельный контроль на территории села Константиновка, а также готовит официальные письма и запросы, подготавливает ответы на запросы физических и юридических лиц. Также в их обязанности входит формирование и предоставление пакета документов в Росреестр и МФЦ для регистрации прав собственности муниципального образования села Константиновка.

Группа, занимающаяся организацией спорта, выполняет широкий спектр обязанностей, включая планирование и проведение разнообразных спортивных мероприятий для людей всех возрастных групп и социальных категорий, начиная от подростков и заканчивая пенсионерами и инвалидами. Они также активно участвуют в организации школьных спартакиад, спортивных вечеров и других мероприятий, направленных на поощрение физической активности и здорового образа жизни. Важным аспектом их работы яв-

ляется обеспечение безопасности участников во время занятий на спортивных площадках и во время проведения соревнований

Организационная структура этого учреждения отличается логической последовательностью, четкими формальными определениями и гибкостью в реализации поставленных задач.

1.2 Структура документооборота

Документооборот – это систематизированный процесс обработки и управления документами в рамках организации или учреждения. Этот процесс включает в себя создание, передачу, хранение, поиск, а также уничтожение документов в соответствии с установленными правилами и процедурами.

Главное правило документооборота состоит в том, что каждый документ должен быть создан, передан, обработан и хранен в соответствии с установленными процедурами и правилами.

1.2.1 Внешний документооборот

Для наглядного представления информационного взаимодействия администрации села Константиновка с физическими и юридическими лицами разработана диаграмма внешнего документооборота, представленная на рисунке 2.

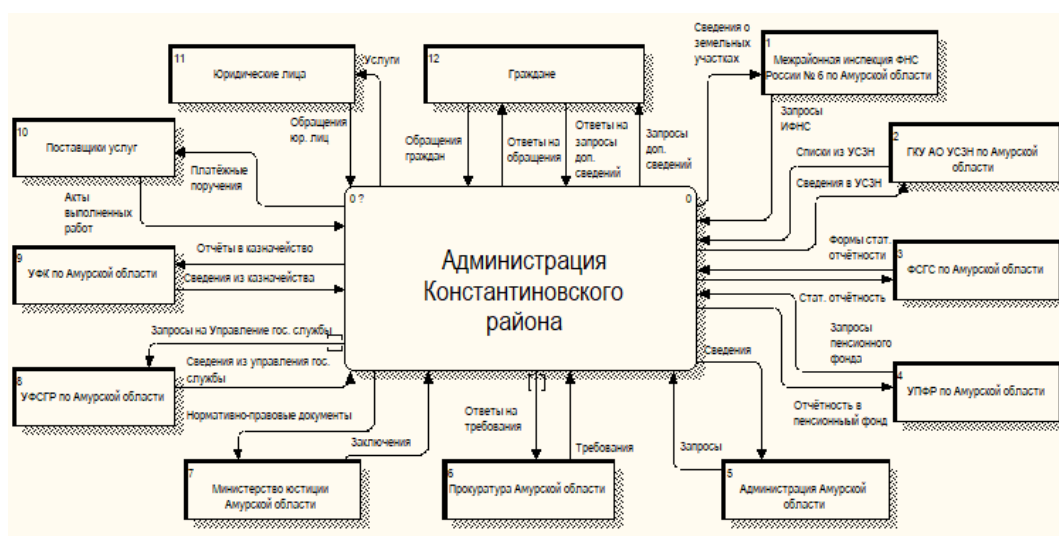


Рисунок 2 – Внешний документооборот администрации Константиновского района

Документооборот в администрации села Константиновка характеризуется следующими потоками информации:

– от Межрайонной ИФНС № 6 по Амурской области к администрации Константиновского района поступают запросы о земельных участках. В то же время, от администрации Константиновского района к Межрайонной ИФНС направляются сведения о земельных участках;

– федеральная служба государственной статистики по Амурской области предоставляет администрации Константиновского района формы статистической отчетности для заполнения. В ответ администрация направляет Отделу статистики различные отчеты, такие как данные о численности населения, о профессиональном обучении муниципальных служащих, о жилищном фонде и другие;

– управление Пенсионного фонда России по Амурской области направляет запросы на сведения о сотрудниках администрации Константиновского района. В свою очередь, администрация Константиновского района предоставляет Пенсионному фонду отчетность;

– правительство Амурской области. Информационным потоком от правительства Амурской области к администрации Константиновского района являются запросы об аварийном жилом фонде, о гражданах, признанных нуждающимися в улучшении жилищных условий. Информационным потоком от администрации Константиновского района к правительству Амурской области является предоставление сведений об аварийном жилом фонде, о гражданах, признанных нуждающимися в улучшении жилищных условий;

– от прокуратуры Амурской области к администрации Константиновского района поступают требования. В ответ администрация направляет Прокуратуре Амурской области ответы на эти требования;

– от Министерства юстиции Амурской области к администрации Константиновского района поступают заключения о соответствии законодатель-

ству нормативно-правовых актов. В ответ администрация направляет Министерству юстиции нормативно-правовые акты на проверку;

– от Управления Федеральной службы государственной регистрации, кадастра и картографии по Амурской области к администрации Константиновского района поступают сведения о недвижимости и информация о земельных участках. В ответ администрация направляет Управлению запросы о наличии недвижимости и о земельных участках у физических лиц;

– от Казначейства Амурской области к администрации Константиновского района поступают данные о выделенных средствах. В ответ администрация направляет Казначейству отчеты о расходовании средств;

– от поставщиков услуг к администрации Константиновского района поступают акты выполненных работ. В ответ администрация направляет поставщикам услуг платежные поручения;

– от юридических лиц к администрации Константиновского района поступают обращения по различным вопросам. В ответ администрация предоставляет услуги по обработке обращений физических лиц;

– граждане обращаются к администрации Константиновского района с запросами о предоставлении муниципальных услуг, информацией, а также ответами на запросы о дополнительных сведениях о гражданах. В ответ администрация предоставляет гражданам ответы на запросы о предоставлении муниципальных услуг и информации, а также запрашивает дополнительные сведения.

1.2.2 Внутренний документооборот

В рамках работы администрации Константиновского района специалисты взаимодействуют в группах, каждая из которых решает свои задачи. Функции различных специалистов в группе пересекаются. Специалист по кадрам, ответственный за делопроизводство, координирует работу между группами, а глава администрации осуществляет контроль за выполнением распоряжений (рис. 3).

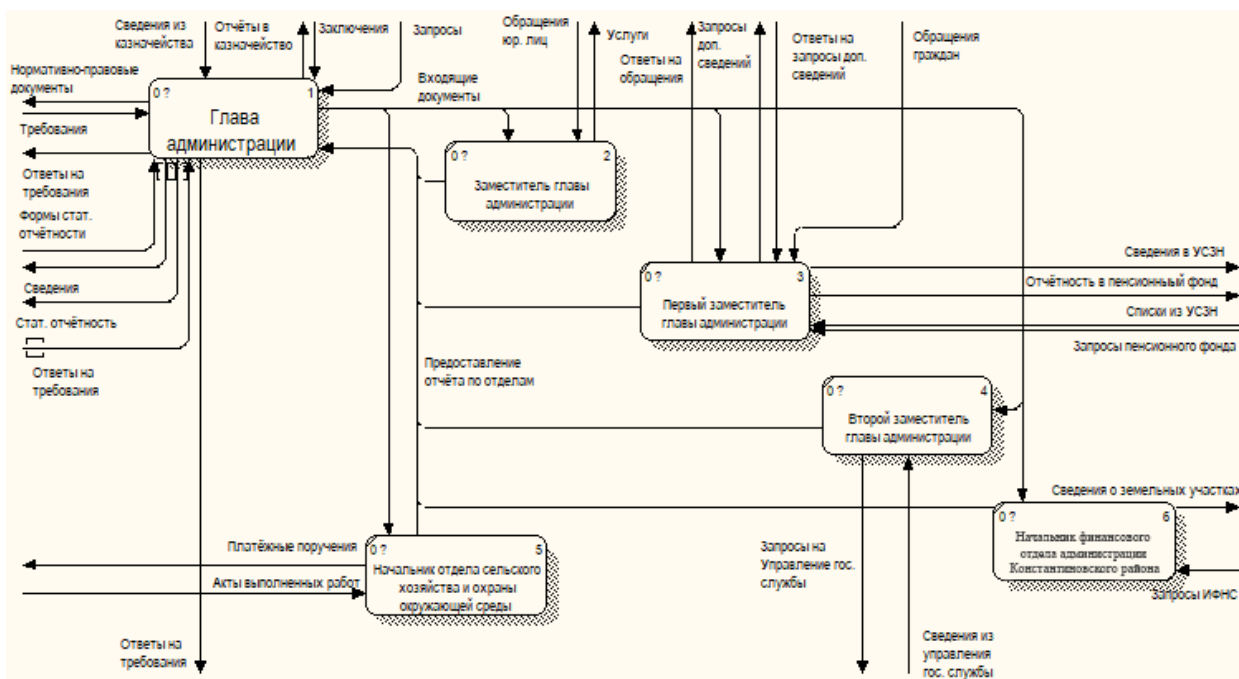


Рисунок 3 – Внутренний документооборот администрации Константиновского района

Глава администрации является главным представителем исполнительной власти в районе. Его обязанности включают руководство администрацией, ее деятельностью по исполнительно-распорядительным функциям. Он также отвечает за формирование состава администрации в соответствии с уставом района и прочими нормативными актами, а также за распределение обязанностей между заместителями в соответствии с уставом и текущими решениями.

Кроме того, глава администрации направляет проекты решений и нормативных актов в районный Совет народных депутатов, что является важным шагом для принятия решений и законов. Ему также поручается исполнение других полномочий в соответствии с федеральными и региональными законами, а также уставом района. Таким образом, глава администрации играет ключевую роль в управлении и организации работы в районе.

Первый заместитель главы администрации района занимает высокий пост в администрации и выполняет свои обязанности в соответствии с полномочиями, установленными уставом района и другими нормативными

правовыми актами. Он принимает временно на себя обязанности главы района в случае его отсутствия или невозможности выполнения обязанностей.

Если как глава, так и первый заместитель не могут выполнить свои функции, исполнение обязанностей главы района может быть передано одному из заместителей главы администрации района, назначенного на должность с согласия районного совета народных депутатов. Таким образом, первый заместитель главы администрации играет важную роль в обеспечении непрерывности работы администрации и эффективного управления районом.

Заместителями главы администрации района являются:

- Заместитель главы администрации района по социальным вопросам, который одновременно является начальником управления образования.
- Заместитель главы администрации района, также занимающий должность начальника финансового отдела.

Компетенция и функции заместителей главы администрации района по координации деятельности соответствующих подразделений устанавливаются главой района. Это включает в себя руководство социальными вопросами и образованием, а также управление финансовыми аспектами деятельности администрации района.

Структурные подразделения администрации района являются ключевыми органами муниципального управления, созданными для эффективного осуществления исполнительной власти. Среди них:

- Аппарат главы района;
- Отраслевые и функциональные подразделения администрации, такие как финансовый отдел и управление образования.

Деятельность этих структурных подразделений администрации района регламентируется соответствующими положениями. Распределение и перераспределение функций и полномочий между ними осуществляется главой района в соответствии с потребностями и приоритетами управления. Это позволяет обеспечить эффективное функционирование администрации и ре-

ализацию задач муниципального управления.

1.3 Пример аналогичной информационной системы

Современные информационные технологии позволяют автоматизировать приемные администрации различных городов, обеспечивая более эффективное и удобное взаимодействие с гражданами. Например, сайт Администрации городского округа города Воронеж внедрил такие решения, которые позволяют жителям получать необходимые услуги и информацию онлайн, минимизируя необходимость личного присутствия и сокращая время на обработку запросов (рис. 4).

Администрация городского округа город Воронеж
ЭЛЕКТРОННАЯ ПРИЕМНАЯ

- Новое обращение
- Отслеживание
- Мой аккаунт
- Правила
- Выход

ТРЕБОВАНИЯ К ТЕКСТУ ОБРАЩЕНИЯ

- В тексте обращения необходимо указать суть обращения (предложения, заявления, жалобы) с указанием адреса места описываемого факта, события, действия.

ТРЕБОВАНИЯ К ПРИЛОЖЕНИЯМ (приложения не являются обязательными)

- Приложения предоставляются в электронной форме **без архивирования**.
- Количество приложений до 5 шт. Суммарный объем не может превышать 10 Мб.
- Допустимы следующие форматы файлов: avi, bmp, doc, docx, flv, gif, jpg, jpeg, mkv, mp3, mp4, mov, odf, odt, psx, pdf, png, ppt, pub, rtf, txt, tif, xls, xlsx, wma, wmv
- Документы, переданные в электронном виде без использования в установленном порядке технологии электронной подписи, не имеют юридической силы. Поэтому такой способ может быть использован только для передачи документов, не требующих удостоверения подлинности документа.

ВНИМАНИЕ! ОБРАЩЕНИЕ НЕ РАССМАТРИВАЕТСЯ, ЕСЛИ:

- в нем содержатся нецензурные либо оскорбительные выражения;
- текст обращения не поддается прочтению;
- обращение не содержит конкретных заявлений, жалоб, предложений;
- ответ по существу поставленного в обращении вопроса не может быть дан без разглашения сведений, составляющих государственную или иную охраняемую законом тайну.

ДОПОЛНИТЕЛЬНАЯ ИНФОРМАЦИЯ

- Поддержка операционной системы **Microsoft Windows XP** прекращена, в связи с окончанием поддержки со стороны производителя и как следствие с невозможностью обеспечения требуемого уровня защищенности передаваемых данных при взаимодействии с сайтом.
- Внимание!** Обращение рассматривается в течение **30 дней** со дня его регистрации в соответствии со статьей 12 Федерального закона № 59-ФЗ «О порядке рассмотрения обращений граждан в Российской Федерации» от 2 мая 2006 года.
- При направлении обращения через «Электронную приемную», ему присваивается идентификационный номер. Информацию о ходе рассмотрения обращения можно получить, воспользовавшись сервисом «[Отслеживание обращений](#)».
- Поступившие через сервис «Электронная приемная» обращения рассматриваются в соответствии с федеральными законами № 59-ФЗ «О порядке рассмотрения обращений граждан в Российской Федерации» от 2 мая 2006 года; № 8-ФЗ «Об обеспечении доступа к информации о деятельности государственных органов и органов местного самоуправления» от 9 февраля 2009 года.
- При направлении обращений, касающихся обжалования судебных решений, необходимо иметь в виду следующее:

Рисунок 4 – «Электронная приемная» администрации округа Воронеж

На данном сайте предоставлена возможность удобно оставлять обращения, которые могут быть как групповыми, так и индивидуальными. Пользователи могут легко заполнить форму обращения, указав все необходимые

данные и описав свою проблему или предложение. Групповые обращения позволяют нескольким гражданам одновременно выразить свое мнение или поднять важный вопрос, что повышает значимость и срочность рассмотрения таких заявлений. Индивидуальные обращения дают возможность каждому жителю напрямую обратиться к администрации по личным вопросам, получать своевременные ответы и решения. Благодаря таким функциональным возможностям, сайт становится эффективным инструментом взаимодействия между гражданами и муниципальными властями, повышая прозрачность и оперативность работы администрации (рис. 5).

Шаг-1 ВВЕДИТЕ ДАННЫЕ ОБРАЩЕНИЯ

Краткое содержание (максимум 100 знаков)
Например: Железнодорожный район, ул. Димитрова д. 79 Вывоз ТБО

Железнодорожный район, ул. Димитрова д. 79 Вывоз ТБО

Текст обращения

Это обращение коллективное

Шаг-2 ДОБАВЬТЕ ПРИЛОЖЕНИЯ (необязательно)

Приложения: 0 из 5, общий объем 0 Мб из 10 Мб
При необходимости Вы можете прикрепить следующие типы файлов: avi, bmp, doc, docx, flv, gif, jpg, jpeg, mkv, mp3, mp4, mov, odf, odt, pcx, pdf, png, ppt, pub, rtf, txt, tif, xls,xlsx, wma, wmv
Общим объемом до 10 Мб. Имя файла может содержать только буквы, цифры и знаки(пробел, нижнее подчеркивание, тире)

[Добавить приложение](#)

Введите цифры с картинки:
3 24 7 7 9

[Изменить картинку](#)

[Отправить](#)

Рисунок 5 – Форма обращения

Кроме того, на сайте доступна сводка правил, подробно описывающая требования к тексту обращений, приложенным документам и дополнительной информации. Это помогает пользователям правильно оформлять свои обращения, что способствует более быстрому и качественному рассмотрению их запросов. Информативность и структурированность данных разделов

делают взаимодействие с администрацией максимально удобным и эффективным, способствуя улучшению качества предоставляемых услуг и удовлетворению потребностей граждан.

Один из примеров электронной приёмной — это сайт администрации города Благовещенка (рис. 6).

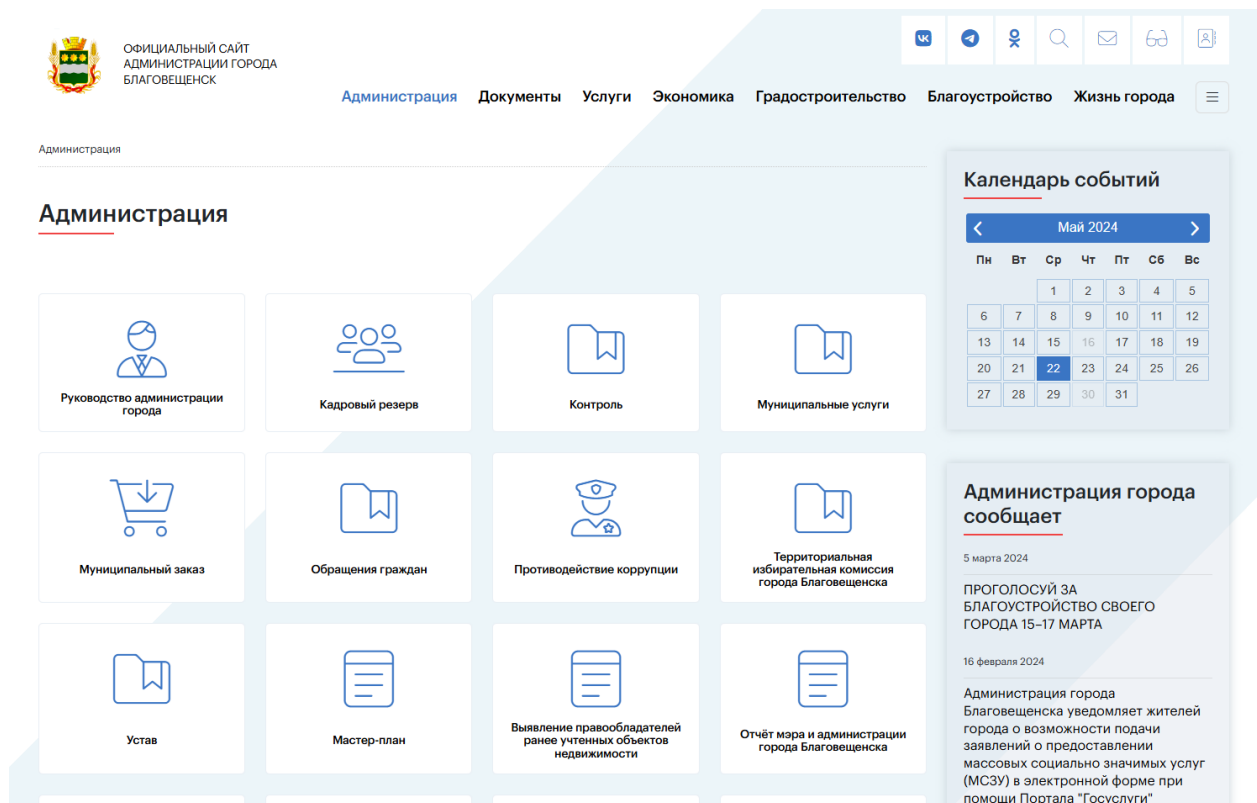


Рисунок 6 – Сайт администрации города Благовещенск

Сайт имеет множество информационных вкладок, каждая из которых предоставляет пользователям доступ к различным важным аспектам городской жизни. Например, в некоторых вкладках можно принять участие в голосованиях города, что позволяет жителям активно участвовать в принятии решений, касающихся их сообщества. В таких голосованиях можно выразить своё мнение по вопросам благоустройства, строительства новых объектов и других значимых тем.

Другие вкладки позволяют отследить бюджет города, предоставляя прозрачную информацию о том, как распределяются и расходуются городские средства. Здесь можно увидеть подробные отчёты о доходах и расходах,

а также узнать о планируемых проектах и их финансировании. Это способствует повышению открытости и подотчётности муниципальных властей перед гражданами.

Кроме того, сайт предоставляет возможность сделать обращение к администрации (рис. 7). Это удобный способ для горожан задать вопросы, выразить свои предложения или жалобы напрямую властям. Такое взаимодействие способствует улучшению коммуникации между жителями и городской администрацией, а также помогает быстрее решать возникающие проблемы.

The image shows a web form titled "Задай вопрос администрации города" (Ask a question to the city administration). The form is set against a light blue background with a white diagonal stripe. It contains the following fields and elements:

- ФИО заявителя** (Applicant's full name): A text input field.
- Тип заявителя:** (Applicant type): Radio buttons for "Физическое лицо" (Selected) and "Юридическое лицо" (Legal entity).
- Соавторы:** (Co-authors): A text input field.
- Адрес проживания:** (Residence address): A text input field containing "г.Благовещенск, ул.Ленина, д.133".
- Телефон** (Phone): A text input field.
- Адрес электронной почты:** (Email address): A text input field containing "name@email.ru".
- Текст обращения:** (Text of the request): A large text area.
- Прикрепить файлы:** (Attach files): A file selection interface showing "Выбор файла" (File selection) and "Не выбран ни один файл" (No file selected).
- Настоящим я даю согласие на обработку персональных данных** (I hereby agree to the processing of personal data).
- Отправить** (Send) button.
- Правила подачи обращений** (Rules for submitting requests) button.

Рисунок 7 – Форма обращения

Создание сайта электронной приемной администрации предоставляет множество преимуществ как для граждан, так и для органов власти. Граждане могут обращаться к администрации в любое время благодаря круглосу-

точной доступности сервиса, экономя при этом время и ресурсы, так как нет необходимости посещать офисы лично. Интуитивно понятный интерфейс облегчает процесс подачи заявлений, жалоб и предложений. Прозрачность повышается за счет возможности отслеживания статуса обращений в режиме реального времени и публикации публичных отчетов и статистики, что укрепляет доверие к администрации.

Эффективность работы администрации улучшается благодаря автоматизации процессов обработки заявлений и жалоб, а также централизованному управлению данными, что облегчает их анализ и управление. Быстрая реакция на поступающие обращения и жалобы способствует улучшению качества обслуживания, а получение обратной связи от граждан помогает выявлять актуальные проблемы и принимать обоснованные решения. Электронные документы снижают затраты на бумажную работу, оптимизируя труд сотрудников, которые могут сосредоточиться на решении более сложных задач.

Электронная приемная также учитывает потребности людей с ограниченными возможностями и может быть адаптирована для различных категорий граждан, что делает её доступной для всех. Многоязыковая поддержка расширяет возможности использования сервиса для людей, говорящих на разных языках. В итоге, создание такого сайта не только улучшает взаимодействие между гражданами и органами власти, но и способствует повышению эффективности и прозрачности работы администрации.

2 РАЗРАБОТКА ИНФОРМАЦИОННОЙ СИСТЕМЫ

2.1 Среда разработки

Для разработки информационной системы «Электронная приёмная» была выбрана среда разработки Visual Studio Code, – это бесплатная и открытая среда разработки (IDE), разработанная компанией Microsoft. Она известна своей высокой производительностью, гибкостью и широкой поддержкой различных языков программирования. Однако, у нее также есть несколько минусов, которые могут быть важны для некоторых пользователей.

Плюсы «Visual Studio Code»:

- бесплатный и открытый исходный код: доступен всем без лицензионных платежей и с возможностью внесения изменений;
- многоплатформенный: работает на Windows, macOS и Linux;
- высокая производительность: оптимизирован для быстрого запуска и работы с кодом;
- гибкая система расширений: доступны тысячи расширений, которые добавляют новые функции, поддержку языков и интеграцию с другими инструментами;
- интеллектуальная автодополнения кода: помогает писать код быстрее и точнее, предлагая варианты завершения кода и подсказки по синтаксису;
- отладка: встроенный отладчик позволяет находить и исправлять ошибки в коде;
- поддержка множества языков программирования: изначально поддерживает JavaScript, PHP, TypeScript, Python, Java и многие другие;
- удобный интерфейс: интуитивно понятный и настраиваемый интерфейс;
- интеграция с Git: VS Code имеет встроенную поддержку Git, что упрощает управление версиями кода;
- активное сообщество: VS Code имеет большое и активное сообщество

ство пользователей и разработчиков, которые делятся информацией, решают проблемы и создают новые расширения;

- мощные функции: VS Code предоставляет множество мощных функций, которые делают его отличным выбором для разработки программ любого размера;

Минусы «Visual Studio Code»:

- некоторые расширения могут быть нестабильными: не все расширения хорошо протестированы и могут вызывать проблемы;

- требует ресурсов: для работы VSCode требуется определенное количество оперативной памяти и дискового пространства, особенно при использовании большого количества расширений;

- некоторые пользователи считают интерфейс не таким интуитивным: пользователи, привыкшие к другим средам разработки, могут найти интерфейс «Visual Studio Code» менее удобным;

- некоторые расширения и функции могут быть сложными в настройке для новичков;

«Visual Studio Code» особенно хорошо подходит для:

- веб-разработки: JavaScript, TypeScript, HTML, CSS, React, Vue.js;

- разработки на Python: с помощью расширений, которые добавляют поддержку фреймворков, таких как Django и Flask;

- разработки на C++: хотя у него нет такой мощной поддержки, как у IntelliJ IDEA, но с помощью расширений он становится более удобным;

В целом VS Code – это гибкий, мощный и удобный редактор кода, который отлично подходит как для новичков, так и для опытных разработчиков.

Для сравнения функционала рассмотрим «IntelliJ IDEA»:

IntelliJ IDEA – это мощная и популярная интегрированная среда разработки (IDE), созданная компанией JetBrains, известной своими высококачественными инструментами для разработчиков. Она специализируется на

разработке на Java, но также отлично подходит для других языков, таких как Kotlin, Groovy, Scala, Python, JavaScript, и многих других.

Плюсы:

– бесплатная версия доступна, но для коммерческого использования нужна платная версия;

– идеально подходит для Java и JVM-языков;

– мощный функционал для рефакторинга;

– умные подсказки и автодополнение кода;

– интеграция с различными фреймворками и инструментами;

– отличная отладка и профилирование;

– хороший набор инструментов для тестирования.

Минусы:

– платная версия для коммерческого использования;

– может быть ресурсоемкой, особенно для слабых машин;

– интерфейс может быть сложнее для новичков, чем у «Visual Studio Code»;

– не такая гибкая система расширений, как у «Visual Studio Code».

IntelliJ IDEA – это мощная и функциональная IDE, которая является отличным выбором для разработки на Java и других языках. Она предлагает широкий спектр возможностей, включая рефакторинг, отладку, интеллектуальное автодополнение кода и интеграцию с инструментами.

Для сравнения функционала рассмотрим ещё один пример - Atom:

Atom - это бесплатная и открытая интегрированная среда разработки (IDE), разработанная компанией GitHub. Она известна своей гибкостью, настраиваемостью и дружелюбным интерфейсом, что делает её популярным выбором для начинающих разработчиков.

Открытый исходный код: это позволяет сообществу разработчиков вносить свой вклад в его развитие, делая Atom более гибким и функциональным.

Плюсы Atom:

– гибкость и настраиваемость: Atom позволяет настраивать внешний вид, функциональность и поведение IDE с помощью темы, стилей и настроек. Вы можете изменить практически все, что вам не нравится;

– дружелюбный интерфейс: Atom имеет простой и интуитивно понятный интерфейс, что делает его удобным в использовании, даже для новичков;

– богатая экосистема пакетов: Тысячи пакетов доступны для расширения функциональности, добавления поддержки новых языков и интеграции с другими инструментами;

– встроенная поддержка Git: Atom включает встроенные инструменты для управления версиями кода с помощью Git;

– разработка в реальном времени: Atom позволяет сотрудничать с другими разработчиками в реальном времени, что делает его удобным для командной разработки.

Минусы Atom:

– низкая производительность: Atom может быть медленным, особенно при работе с большими проектами;

– нестабильность: Atom может быть нестабильным и вызывать сбои;

– ограниченные возможности для отладки: Atom не предлагает таких же мощных возможностей для отладки, как IntelliJ IDEA или VS Code;

– меньшая популярность: в сравнении с VS Code и IntelliJ IDEA у Atom меньше сообщество и меньше поддержки.

В итоге Atom – это гибкий и настраиваемый редактор кода, хорошо подходящий для новичков и небольших проектов. Однако, его низкая производительность и нестабильность могут быть проблемой для профессиональных разработчиков и крупных проектов.

2.2 Проектирование баз данных

Проектирование баз данных проходило в три этапа:

- инфологическое проектирование – выделение сущностей и назначение им атрибутов;
- логическое проектирование – построение логической структуры базы данных, приведение отношений к нормальным формам;
- физическое проектирование – описываются таблицы в том виде, в котором они реализованы средствами.

2.2.1 Инфологическое проектирование

Задача инфологического проектирования – это получение семантических моделей, отражающих информационное содержание выбранной ПО. Целью является – обеспечение наиболее естественного для пользователя способов для сбора и представления той информации, которую предполагается хранить в создаваемой БД.

Спецификация сущностей:

На основании проведенных исследований предметной области, составленного технического задания и учета структуры документов были выделены следующие сущности, необходимые для построения информационной базы:

- «Пользователи» – сведения о пользователях электронной приёмной;
- «Обращения» – сведения о созданных обращениях;
- «Кабинеты» – сведения о кабинетах, разбирающие обращения;
- «Модератор» – сведения о модерации сайта;
- «Работник администрации» – сведения о работниках, разбирающих обращения.

Спецификация атрибутов:

Назначим приведенным выше сущностям описательные атрибуты в форме таблиц, представленных ниже:

Атрибуты сущности «Пользователи»

Описание атрибутов сущности «Пользователи» приведено в таблице 1. Для идентификации экземпляра сущности вводим атрибут «Номер», который будет являться первичным ключом для данной сущности.

Таблица 1 – Атрибуты сущности «Пользователи»

Название атрибута	Описание атрибута	Диапазон значений	Единица измерения	Пример значения атрибута
<u>Номер</u>	Уникальный идентификатор	>0	-	2
Имя	Имя пользователя	2<firstname<10	-	Михаил
Фамилия	Фамилия Пользователя	2<secondname<15	-	Комогорцев
Электронная почта	Адрес электронной почты	-	-	Ivan_petrov@mail.ru
Дополнительная информация	Дополнительная информация о пользователях	about_me<1000	-	Я занимаюсь спортом и разработкой сайтов!
Дата рождения	Дата рождения пользователя	>01.01.1900	Гггг. мм. дд.	1988-10-14
Номер ключа доступа	Специальный ключ, присваиваемый администраторам	0	-	1
Пароль	Пароль пользователя	-	-	Qwerty123

Атрибуты сущности «Обращения»

Описание атрибутов сущности «Обращения» приведено в таблице 2. Для идентификации экземпляра сущности вводим атрибут «Номер», который будет являться первичным ключом для данной сущности.

Таблица 2 – Атрибуты сущности «Обращения»

Название атрибута	Описание атрибута	Диапазон значений	Единица измерения	Пример значения атрибута
<u>Номер</u>	Уникальный идентификатор	>0	-	2

Продолжение таблицы 2

Электронный адрес	Адрес электронной почты заявителя	-	-	Ivan_petrov@mail.ru
Описание	Содержит текст обращения	20<description_post<1000	-	Отремонтируйте дороги! Ездить невозможно! Из-за ям пробил колесо!
Причина	Причина обращения	reason_post<20	-	Проблема с дорогами!
Дата	Дата обращения	>01.01.1900	Гггг. мм. дд.	1988-10-14
Статус	Текущий статус обращения	-	-	Новое
Дата записи	Дата записи	>01.01.1900	Гггг.мм. дд.	2024-05-23
Пояснение	Пояснение для пользователя	-	-	Подойдите к бухгалтеру

Атрибуты сущности «Кабинеты»

Описание атрибутов сущности «Кабинеты» приведено в таблице 3. Для идентификации экземпляра сущности вводим атрибут «Номер», который будет являться первичным ключом для данной сущности.

Название атрибута	Описание атрибута	Диапазон значений	Единица измерения	Пример значения атрибута
<u>Номер</u>	Уникальный идентификатор	>0	-	2
Номер кабинета	Номер кабинета	-	-	123
Заведующий	ФИО заведующего кабинетом	-	-	Михаил Евгеньевич Павлов
Вид деятельности	Вид деятельности	-	-	Бухгалтерия

Атрибуты сущности «Модератор»

Описание атрибутов сущности «Модератор» приведено в таблице 4. Для идентификации экземпляра сущности вводим атрибут «Номер», который будет являться первичным ключом для данной сущности.

Таблица 4 – Атрибуты сущности «Обращения»

Название атрибута	Описание атрибута	Диапазон значений	Единица измерения	Пример значения атрибута
<u>Номер</u>	Уникальный идентификатор	>0	-	3
Имя	Имя модератора	2<firstname<10	-	Алексей
Фамилия	Фамилия модератора	2<secondname<15	-	Кравцев
Электронная почта	Адрес электронной почты	-	-	Petrov_123@mail.ru
Дополнительная информация	Дополнительная информация о модераторе	about_me<1000	-	Я модерацией сайт
Дата рождения	Дата рождения модератора	>01.01.1900	Гггг. мм. дд.	1988-10-14
Номер ключа доступа	Специальный ключ, присваиваемый администраторам	1	-	1
Пароль	Пароль пользователя	-	-	Qwerty123

Атрибуты сущности «Работник администрации»

Описание атрибутов сущности «Работник администрации» приведено в таблице 5. Для идентификации экземпляра сущности вводим атрибут «Номер», который будет являться первичным ключом для данной сущности.

Таблица 5 – Атрибуты сущности «Пользователи»

Название атрибута	Описание атрибута	Диапазон значений	Единица измерения	Пример значения атрибута
<u>Номер</u>	Уникальный идентификатор	>0	-	4
Имя	Имя работника администрации	2<firstname<10	-	Виталий
Фамилия	Фамилия работника администрации	2<secondname<15	-	Зави-юшкин
Электронная почта	Адрес электронной почты	-	-	lokin_1257@mail.ru
Дополнительная информация	Дополнительная информация о работнике администрации	about_me<1000	-	Я работник администрации и выполняю свои обязанности
Дата рождения	Дата рождения работника администрации	>01.01.1900	Гггг. мм. дд.	1988-10-14
Номер ключа доступа	Специальный ключ присваиваемый администраторам	2	-	1
Пароль	Пароль работника администрации	-	-	Qwer-ty123

Определение связей между сущностями

Охарактеризуем связи, установленные между сущностями:

Связь «Пользователи» – «Обращения» имеет связь один-ко-многим, так как один пользователь может иметь несколько обращений, а каждое обращение принадлежит может относиться только к одному пользователю (рис. 8).



Рисунок 8 – Отношение «Пользователи» – «Обращения»

Связи «Обращения» – «Кабинеты» имеет связь многие-к-одному, Так как множество обращений могут быть определены в один кабинет, но несколько кабинетов не могут рассматривать одно обращение.



Рисунок 9 – Отношение «Обращения» – «Кабинеты»

Связь «Работники администрации» – «Кабинеты» имеет связь многие-ко-многим, так как несколько работников могут работать с одним кабинетом, и несколько кабинетов могут относиться к одному сотруднику.



Рисунок 10 – Отношение «Работники администрации» – «Кабинеты»

Связь «Работники администрации» - «Обращения» имеет связь многие-ко-многим, так как один сотрудник может обрабатывать несколько обращений, и одно обращение могут обрабатывать несколько сотрудников.



Рисунок 11 – Отношение «Работники администрации» – «Обращения»

Связь «Модераторы» - «Обращения» имеет связь многие-ко-многим, так как один модератор может обрабатывать несколько обращений, и одно обращение могут обрабатывать несколько модераторов.



Рисунок 12 – Отношение «Модераторы» – «Обращения»

2.2.2 Логическое проектирование

Цель данного этапа состоит в построении реляционной логической модели. Реляционная логическая модель представляет собой совокупность нормализованных отношений, в которых реализованы связи между объектами предметной области и выполнены все преобразования, необходимые для ее эффективной реализации в среде конкретной СУБД.

Для получения такой модели базы данных необходимо:

- отобразить полученную модель на реляционную путём совместного представления ключевых элементов взаимосвязанных сущностей;
- выполнить анализ полученных отношений на соответствие их требованиям трех нормальных форм;
- представить полученную реляционную логическую модель.

Таким образом, реляционная логическая модель обеспечивает структурированное и эффективное представление данных, что важно для обеспечения целостности, надёжности и производительности системы управления базами данных (СУБД).

Графическое представление концептуальной инфологической модели базы данных «Электронная приёмная» представлена на рисунке 12.

На данной модели представлены все сущности исследованной предметной области и определены связи между ними.

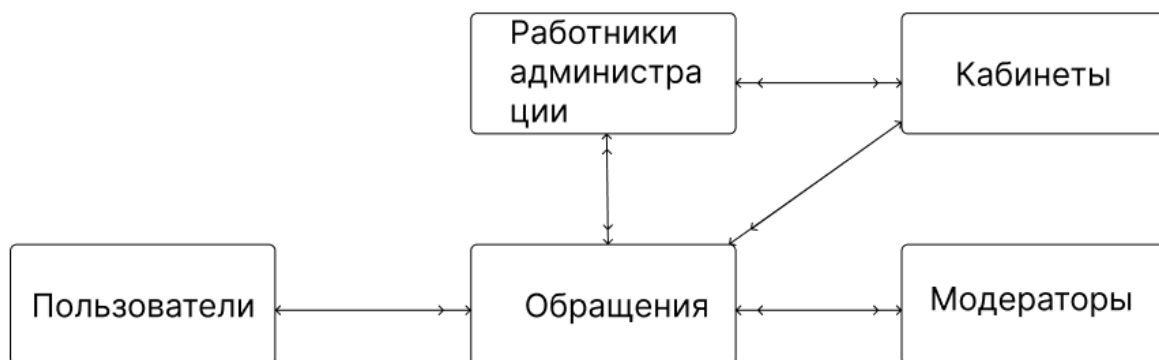


Рисунок 12 – Инфологическая модель базы данных

Пользователи:

<u>Номер</u>	Имя	Фамилия	Почта	О себе	Дата рожд	Ключ доступа	Пароль
--------------	-----	---------	-------	--------	-----------	--------------	--------

Рисунок 13 – Модель «Пользователи»

Модераторы:

<u>Номер</u>	Имя	Фамилия	Почта	О себе	Дата рожд	Ключ доступа	Пароль
--------------	-----	---------	-------	--------	-----------	--------------	--------

Рисунок 14 – Модель «Модераторы»

Работник администрации:

<u>Номер</u>	Имя	Фамилия	Почта	О себе	Дата рожд	Ключ доступа	Пароль
--------------	-----	---------	-------	--------	-----------	--------------	--------

Рисунок 15 – Модель «Работник администрации»

Обращения:

<u>Номер</u>	Почта	Описание	Причина	Дата	Статус
--------------	-------	----------	---------	------	--------

Рисунок 16 – Модель «Обращения»

Кабинеты:

<u>Номер</u>	Номер каб	Заведующ	Вид актив
--------------	-----------	----------	-----------

Рисунок 17 – Модель «Кабинеты»

Отображение инфологической модели на реляционную модель выполняются путём совместного представления ключевых элементов взаимосвязанных сущностей.

Сущности «Пользователи» и «Обращения». Между ними установлена связь типа «один ко многим».



Рисунок 18 – Связь «Пользователи – Обращения»

Поскольку простая связь исходит от сущности «Пользователи», то данная сущность будет исходной, соответственно сущность «Обращения» будет порождённой.

Сущности «Обращения» и «Кабинеты». Между ними установлена связь типа «один ко многим».



Рисунок 19 – Связь «Обращения – Кабинеты»

Поскольку простая связь исходит от сущности «Кабинеты», то данная сущность будет исходной, соответственно сущность «Обращения» будет порождённой. От исходной сущности необходимо добавить «Номер кабинета» (рис. 20).

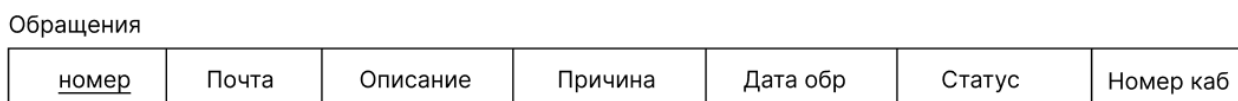


Рисунок 20 – «Сущность обращения»

Сущности «Работники администрации» и «Обращения». Между ними установлена связь типа «многие ко многим».



Рисунок 21 – Связь «Работники администрации – Обращения»

Сущности «Модератор» и «Обращения». Между ними установлена связь типа «многие ко многим».



Рисунок 22 – Связь «Модератор – Обращения»

Сущности «Работник администрации» и «Кабинеты». Между ними установлена связь типа «многие ко многим».



Рисунок 23 – Связь «Работник администрации – Кабинеты»

Получаем итоговый набор реляционных отношений базы данных, в которых исключено дублирование в сущностях, т.е. совместное представление ключей взаимосвязанных сущностей. Далее необходимо нормализовать отношения.

Нормализация отношений – это формальный аппарат ограничений на формирование отношений, который позволяет устранить дублирование, обеспечивает непротиворечивость хранимых данных и уменьшает трудоза-

траты на ведение базы данных.

Отношение называется нормализованным или приведённым к первой нормальной форме, если все его атрибуты простые, т.е. значение атрибутов не является множеством или повторяющейся группой.

Отношение «Пользователи» соответствует второй нормальной форме.



Рисунок 24 – Отношения «Пользователи»

Отношение «Пользователи» отвечает требованиям 2НФ, т.к. оно соответствует 1НФ, и каждый не ключевой атрибут в этом отношении полностью зависим от ключа.

Отношение «Обращения» соответствует второй нормальной форме.



Рисунок 25 – Отношения «Пользователи»

Отношение «Обращения» отвечает требованиям 2НФ, т.к. оно соответствует 1НФ, и каждый не ключевой атрибут в этом отношении полностью зависим от ключа.

Отношение «Кабинет» соответствует второй нормальной форме.



Рисунок 26 – Отношения «Кабинеты»

Отношение «Кабинет» отвечает требованиям 2НФ, т.к. оно соответствует 1НФ, и каждый не ключевой атрибут в этом отношении полностью зависим от ключа.

Отношение «Модератор» соответствует второй нормальной форме.



Рисунок 27 – Отношения «Модератор»

Отношение «Модератор» отвечает требованиям 2НФ, т.к. оно соответствует 1НФ, и каждый не ключевой атрибут в этом отношении полностью зависит от ключа.

Отношение «Работник администрации» соответствует второй нормальной форме.



Рисунок 28 – Отношения «Кабинеты»

Отношение «Работник администрации» отвечает требованиям 2НФ, т.к. оно соответствует 1НФ, и каждый не ключевой атрибут в этом отношении полностью зависит от ключа.

Все полученные нами на предыдущем этапе отношения находятся во второй нормальной форме, и каждый не ключевой атрибут не транзитивно зависит от ключа. Так как между атрибутами оставшихся отношений нет транзитивной зависимости, то, следовательно, эти отношения соответствуют требованиям 3НФ, и дальнейшей нормализации не требуется.

В результате получается итоговая логическая модель базы данных.

Нотация Чена изображена на рисунке 29

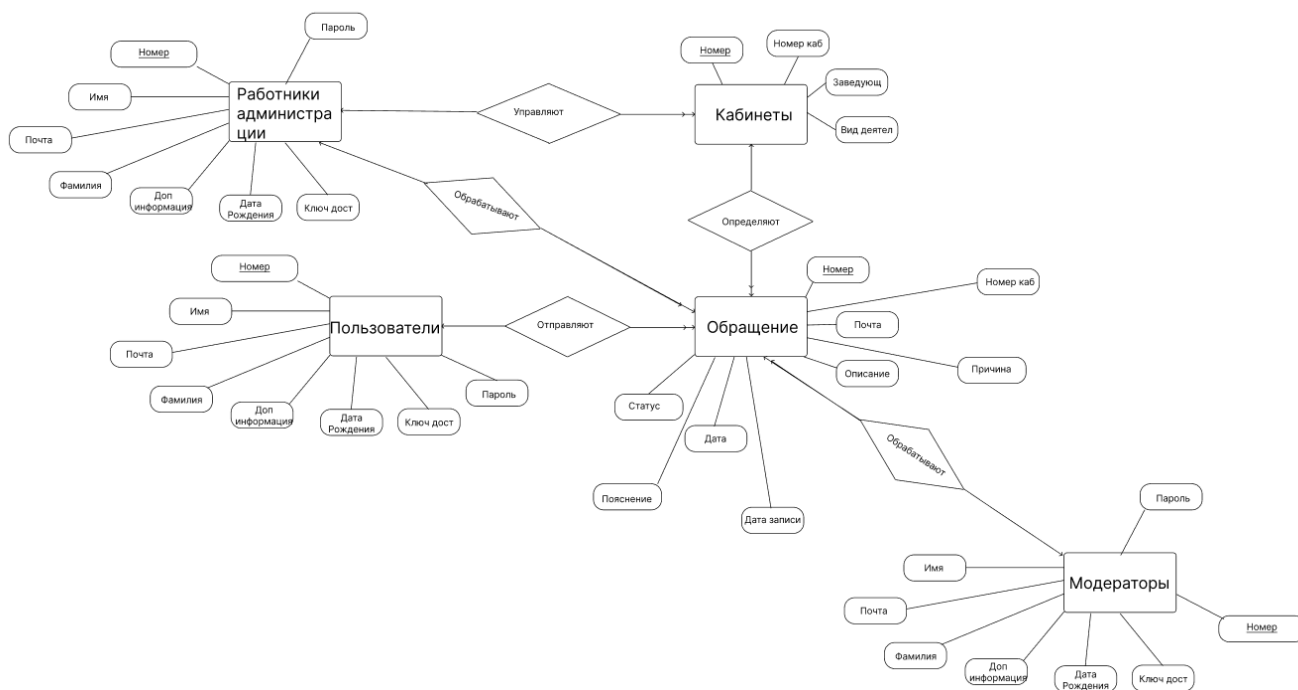


Рисунок 29 – Нотация Чена

2.2.3 Физическое проектирование

На этом этапе проектирования базы данных составлены проекты таблиц, которые будут реализованы средствами СУБД phpMyAdmin. Имена таблиц и их полей приведены в таблицах 6-10.

Таблица 6 – Проект таблицы «Пользователи»

Наименование поля	Тип данных	Ограничения	Значение по умолчанию	Допустимость неопределенного значения	Индексация
<u>Номер</u>	INT	>0	-	Is Not Null	Да
Имя	VARCHAR(191)	-	-	Is Not Null	Нет
Фамилия	VARCHAR(191)	-	-	Is Not Null	Нет
Электронная почта	VARCHAR(191)	-	-	Is Not Null	нет
Дополни-	VARCHAR	<1000	-	Is Not Null	нет

Продолжение таблицы 6

тельная информация	R(191)				
Дата рождения	DATE	-	-	Is Not Null	нет
Номер ключа доступа	INT	-	0	Is Not Null	нет
Пароль	VARCHAR(191)	-	-	Is Not Null	нет

Таблица 7 – Проект таблицы «Модераторы»

Наименование поля	Тип данных	Ограничения	Значение по умолчанию	Допустимость неопределенного значения	Индексация
<u>Номер</u>	INT	>0	-	Is Not Null	Да
Имя	VARCHAR(191)	-	-	Is Not Null	Нет
Фамилия	VARCHAR(191)	-	-	Is Not Null	Нет
Электронная почта	VARCHAR(191)	-	-	Is Not Null	нет
Дополнительная информация	VARCHAR(191)	<1000		Is Not Null	нет
Дата рождения	DATE	-	-	Is Not Null	нет
Номер ключа доступа	INT	-	0	Is Not Null	нет
Пароль	VARCHAR(191)	-	-	Is Not Null	нет

Таблица 8 – Проект таблицы «Работник администрации»

Наименование поля	Тип данных	Ограничения	Значение по умолчанию	Допустимость неопределенного значения	Индексация
<u>Номер</u>	INT	>0	-	Is Not Null	Да
Имя	VARCHAR(191)	-	-	Is Not Null	Нет
Фамилия	VARCHAR(191)	-	-	Is Not Null	Нет
Электронная почта	VARCHAR(191)	-	-	Is Not Null	нет
Дополнительная информация	VARCHAR(191)	<1000		Is Not Null	нет
Дата рождения	DATE	-	-	Is Not Null	нет
Номер ключа доступа	INT	-	0	Is Not Null	нет
Пароль	VARCHAR(191)	-	-	Is Not Null	нет

Таблица 9 – Проект таблицы «Обращения»

Наименование поля	Тип данных	Ограничения	Значение по умолчанию	Допустимость неопределенного значения	Индексация
<u>Номер</u>	INT	>0	-	Is Not Null	Да
Электронный адрес	VARCHAR(191)	-	-	Is Not Null	Нет
Описание	VARCHAR(191)	<1000	-	Is Not Null	Нет
Причина	VARCHAR(191)	-	-	Is Not Null	нет
Дата	DATE	-		Is Not Null	нет
Статус	VARCHAR(191)	-	-	Is Not Null	нет

Продолжение таблицы 9

Номер кабинета	INT	>0	-	Is Not Null	нет
Дата записи	DATE	-	-	Is Not Null	нет
Пояснение	VARCHAR(191)	-	-	Is Not Null	нет

Таблица 10 – Проект таблицы «Кабинеты»

Наименование поля	Тип данных	Ограничения	Значение по умолчанию	Допустимость неопределенного значения	Индексация
<u>Номер</u>	INT	-	-	Is Not Null	Нет
Номер кабинета	VARCHAR(191)	<100 0	-	Is Not Null	Нет
Заведующий	VARCHAR(191)	-	-	Is Not Null	нет
Вид деятельности	VARCHAR(191)	-	-	Is Not Null	нет

2.4 Разработка сайта

Постановка целей и задач:

Для начала разработки сайта было необходимо определить, как он будет функционировать и какие задачи выполнять, для этого был проведён анализ логического и физического проектирования базы данных, описания предметной области, а также составлена нотация Чена.

В ходе анализа были поставлены следующие цели и задачи:

Цели:

- интерфейс должен быть интуитивно понятен;
- должна присутствовать адаптация под различные устройства;
- помимо основного функционала, должна быть дополнительная информация.

Задачи:

- реализовать регистрацию и авторизацию;
- реализовать личный кабинет с возможностью изменения данных;
- реализовать панель управления;
- реализовать меню создания обращений.

Вёрстка сайта:

На данном этапе разрабатывается интерфейс, видимый пользователю и администратору сайта. Разработка дизайна проводилась одновременно с вёрсткой сайта, что позволило свободно выбирать расположение различных блоков и их вид. В процессе разработки использовались дополнительные библиотеки, ускоряющие и упрощающие процесс разработки, такие как Bootstrap версии 5.0 и jQuery.

Итоговый вид главной страницы представлен на рисунке 30.

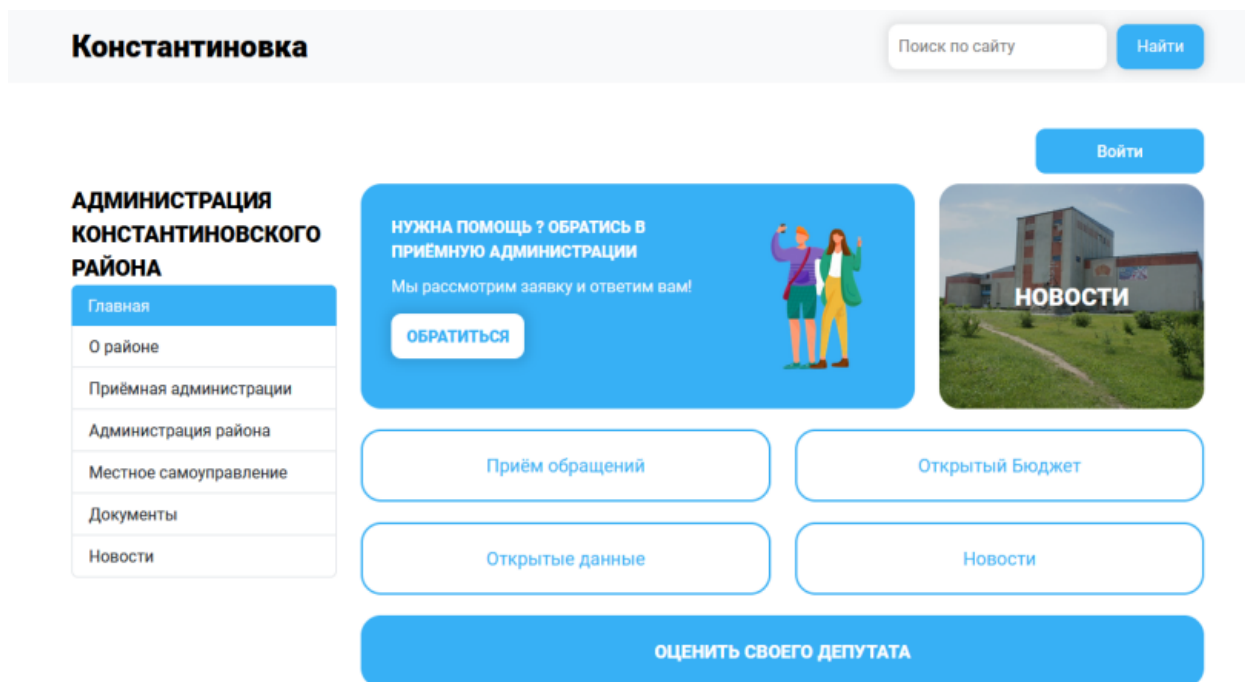


Рисунок 30 – Главная страница

Для интерфейса главной страницы сайта было необходимо добавить информацию о районе и администрации. С этой целью было разработано бо-

ковое меню, позволяющее пользователям просматривать информацию о районе или обращаться к администрации. Меню интуитивно понятное и при нажатии изменяет содержимое страницы в соответствии с потребностями пользователя. Для реализации этого меню использовался JavaScript, который сопоставляет идентификаторы блоков с идентификаторами бокового меню, скрывает ненужные блоки и отображает активный блок перед пользователем.

Все страницы были разработаны с использованием библиотеки Bootstrap версии 5.0. В процессе разработки применялись встроенные классы этой библиотеки, такие как `container`, `row` и `col`.

`Container` – это основной строительный блок Bootstrap, который содержит, дополняет и выравнивает контент в рамках определённого устройства или области просмотра. Контейнеры являются ключевым элементом макета в Bootstrap и необходимы при использовании системы `grid` по умолчанию. Они используются для размещения, разметки и центрирования содержимого внутри себя.

`Row` – это гибкая мобильная сетка `flexbox-grid`, которая позволяет создавать макеты любых форм и размеров благодаря системе из двенадцати столбцов и шести адаптивных уровней по умолчанию. `Flexbox-grid` использует контейнеры, строки и столбцы для компоновки и выравнивания содержимого, она создана с использованием `flexbox` и полностью адаптивна.

`Col` – столбцы создаются на основе архитектуры `grid flexbox`. С помощью `Flexbox` у вас есть возможность изменять отдельные столбцы и группы столбцов на уровне строк. Вы сами решаете, как увеличить, уменьшить или изменить столбцы другим способом.

При создании макетов сетки всё содержимое размещается в столбцах. Иерархия сетки Bootstrap начинается с контейнера, продолжается строкой, затем столбцом и заканчивается содержимым.

Серверная часть:

Для разработки серверной части был использован язык программирования PHP и библиотека RedBeanPHP. Эта библиотека была создана в 2009 году и считается достаточно зрелой. С 2013 года не было обнаружено серьезных ошибок, а в последние годы были добавлены только незначительные функции. Кодовая база тестируется с каждым изменением, и проведено более 20 338 модульных тестов (100 % покрытия кода). Проект активно поддерживается, и разработчики серьезно относятся к обратной связи.

Для реализации авторизации и регистрации была создана общая база данных в phpMyAdmin (рис. 31).

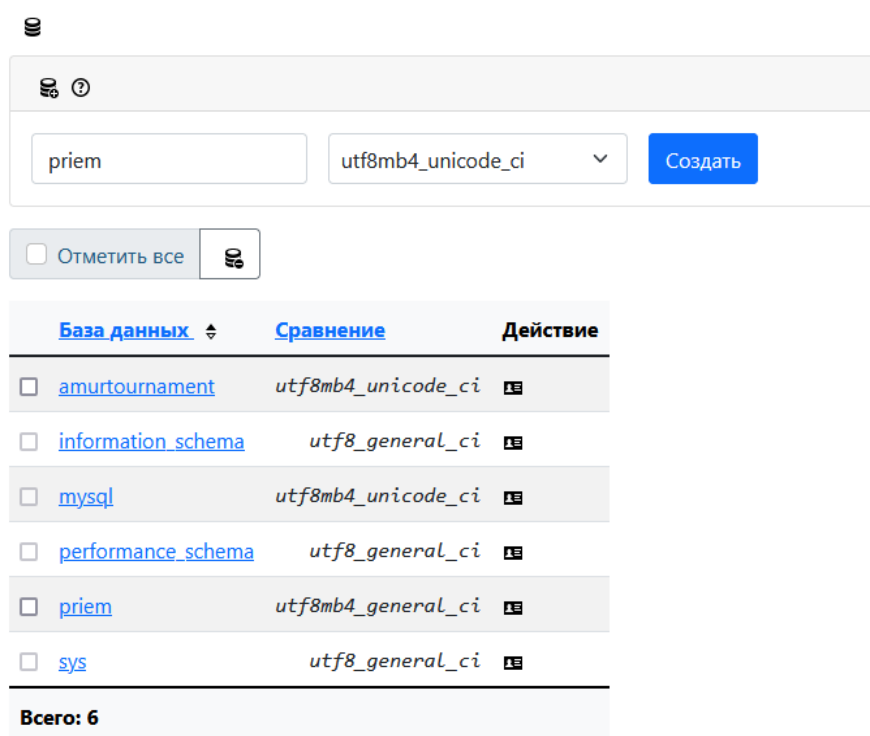


Рисунок 31 – Создание базы данных в phpMyAdmin

В дальнейшем использование phpMyAdmin не потребуется и все манипуляции с базой данных будут выполняться с помощью RedBeanPHP.

Чтобы подключить проект к базе данных необходимо выполнить соединение, данная реализация вынесена в отдельный файл (рис. 32) и используется в остальных при необходимости с помощью следующей команды: `require $_SERVER['DOCUMENT_ROOT'] . '/includes/connect_db.php';`

```

includes > connect_db.php
1  <?
2
3  session_start();
4
5  require $_SERVER['DOCUMENT_ROOT'] . '/rb/rb.php';
6
7  R::setup("mysql:host=127.0.0.1;dbname=priem;", "root", "", false);
8
9  if (!R::testConnection()) die('No DB connection!');

```

Рисунок 32 – Подключение к базе данных

Для реализации регистрации и её корректной работы необходимо создать валидацию вводимых данных. Создаём пустой массив \$errors, в который будем помещать все ошибки. Для поля «Имя» необходимо, чтобы все символы соответствовали кириллице, а также длина имени была больше 2, но меньше 10 символов. Валидация представлена на рисунке 33.

```

if(!preg_match('/[А-Яа-я]/', $_POST['firstname']))
{
    array_push($errors, 'Введите корректное имя');
}
if(strlen($_POST['firstname']) < 2)
{
    array_push($errors, 'Длина имени не может быть меньше 2 символов');
}
if(strlen($_POST['firstname']) > 10)
{
    array_push($errors, 'Длина имени не может превышать 10 символов');
}

```

Рисунок 33 – Валидация поля «Имя»

На рисунке 33 представлена диаграмма или схема, иллюстрирующая процесс валидации данных для регистрации. Сначала пользователь вводит данные в форму. Затем данные отправляются на сервер, где проходит этап валидации. Если данные не соответствуют установленным критериям, ошибки добавляются в массив \$errors и возвращаются пользователю для исправ-

ления. Если данные корректны, процесс регистрации продолжается, и информация сохраняется в базе данных.

Для поля «Фамилия» необходимо, чтобы все символы соответствовали кириллице, а также длина фамилии была больше 2, но меньше 25 символов. Валидация представлена на рисунке 34.

```
if(!preg_match('/[А-Яа-я]/', $_POST['secondname']))
{
    array_push($errors, 'Введите корректную фамилию');
}
if(mb_strlen($_POST['secondname'], 'UTF-8') < 2)
{
    array_push($errors, 'Длина фамилии не может быть меньше 2 символов');
}
if(mb_strlen($_POST['secondname'], 'UTF-8') > 25)
{
    array_push($errors, 'Длина фамилии не может превышать 25 символов');
}
```

Рисунок 34 – Валидация поля «Фамилия»

В валидации полей «Имя» и «Фамилия» использовался метод `mb_strlen` с параметром 'UTF-8' для перевода введённых символов в нужную кодировку, иначе символы кириллицы будут считаться за 2 байта. Метод `array_push` помещает в массив `$errors` текст ошибки.

Для поля «Email» необходимо ввести адрес по форме, для этого используется метод `filter_var` с параметром `FILTER_VALIDATE_EMAIL` который проверяет корректно ли введён адрес почты (рис. 35).

```
if(!filter_var($_POST['email'], FILTER_VALIDATE_EMAIL))
{
    array_push($errors, 'Введите корректный email');
}
```

Рисунок 35 – Валидация адреса электронной почты

Валидация поля «Пароль» и «Подтверждение пароля» заключается в проверке на количество символов, то есть пароль должен быть больше 6, но

меньше 20 символов и оба поля с введенным паролем должны совпадать (рис. 36).

```
if(trim($_POST['password']) == ''){
    array_push($errors, 'Укажите пароль');
}
if(strlen($_POST['password']) < 6)
{
    array_push($errors, 'Длина пароля должна превышать 6 символов');
}
if(strlen($_POST['password']) > 20)
{
    array_push($errors, 'Длина пароля не может превышать 20 символов');
}
if($_POST['confirm_password'] != $_POST['password']){
    array_push($errors, 'Пароли не совпадают');
}
```

Рисунок 36 – Валидация поля «Пароль» и «Подтверждение пароля»

Далее необходимо проверить имеется ли пользователь с таким адресом электронной почты в базе данных или нет. В данной валидации используется метод RedBeanPHP «R::count» который считает количество пользователей в таблице «users» с данной почтой. Если хоть один такой пользователь найдётся, то в массив добавится ошибка (рис. 37).

```
if(R::count('users', 'email = ?', array($_POST['email']))>0){
    array_push($errors, 'Пользователь с таким email уже зарегистрирован!');
}
```

Рисунок 37 – Проверка пользователя

Если после всех валидаций не было найдено ошибок и массив остался пустым, то происходит сохранение пользователя, а также присваивается «key_id» равным 0. Метод password_hash с параметром PASSWORD_DEFAULT необходим для того, чтобы в базе данных хранился не сам пароль, а его хеш в формате bcrypt. На данный момент это является самым надёжным способом хранения пароля. После сохранения данных о пользователе происходит перенаправление на страницу авторизации (рис. 38).

```

if(!empty($errors)){
    echo $errors[0];
}
else{
    $user = R::dispense('users');
    $user->key_id = $_POST[false];
    $user->firstname = $_POST['firstname'];
    $user->secondname = $_POST['secondname'];
    $user->email = $_POST['email'];
    $user->password = password_hash( $_POST['password'], PASSWORD_DEFAULT );
    R::store($user);
    header('Location: auth.php');
}

```

Рисунок 38 – Внесение данных в базу данных

На странице авторизации производим аналогичные действия. Находим пользователя в базе данных с помощью метода `R::findOne` и передаём ему в качестве параметра введённый пользователем адрес электронной почты. Если данный пользователь был найден, то сверяем его хеш пароля с введённым паролем (рис. 39).

```

$user = R::findOne('users', 'email = ?', [$_POST['email']]);
if (empty($user)) {
    array_push($errors, 'Пользователь не найден');
}
if (password_verify($_POST['password'], $user['password'])) {
} else {
    array_push($errors, 'Неверный пароль');
}

if (!empty($errors)) {
} else {
    $user = $user->export();
    $_SESSION['user_data'] = $user;
    header('Location: /');
}

```

Рисунок 39 – Авторизация

Если ошибок не было найдено, то данные пользователя из базы данных преобразуются в массив и присваиваются текущей сессии. После успешной авторизации пользователя перенаправляет на главную страницу, с обновлённой кнопкой «Личный кабинет» вместо «Войти» (рис. 40).

```

<div class="col-2 text-center">
  <? if ($_SESSION['user_data']['key_id'] == true): ?>
    <a href="/management.php">
      <div class="user_edit_profile">
        Админ панель
      </div>
    </a>
  <? endif; ?>
</div>
<div class="col-2 text-center">
  <? if ($_SESSION['user_data']): ?>
    <a href="/edit_profile.php">
      <div class="user_edit_profile">
        Личный кабинет
      </div>
    </a>
  <? else: ?>
    <a href="/auth.php">
      <div class="user_edit_profile">
        Войти
      </div>
    </a>
  <? endif; ?>
</div>

```

Рисунок 40 – Смена кнопок в зависимости от сессии

Если же пользователю присвоен `key_id = 1`, то на главной странице появится ещё и кнопка «Админ панель».

Далее необходимо было реализовать личный кабинет и создание обращений. Для этого было решено использовать AJAX (аббревиатура от «Asynchronous Javascript And Xml») – технология обращения к серверу без перезагрузки страницы.

За счёт этого уменьшается время отклика и веб-приложение по интерактивности больше напоминает десктоп. Несмотря на то, что в названии технологии присутствует буква `x` (от слова XML), использовать XML вовсе не обязательно. Под AJAX подразумевают любое общение с сервером без перезагрузки страницы, организованное при помощи JavaScript.

При открытии личного кабинета пользователь попадает в первый блок, в котором на данный момент уже отображены введённые им данные при регистрации, а также имеется возможность установить фото профиля. Это реализовано с помощью передачи в `input` данных пользователя (рис.41).

```

<div class="col-9 text-start">
  <div class="row row_corrector_edit_menu">
    <div class="col-12">
      <p class="text_account">Имя</p>
      <input id="input_firstname" class="input_account"
        value="<? echo $_SESSION['user_data']['firstname'] ?>" type="text">
      <div class="decoration_line"></div>
      <p class="text_account">Фамилия</p>
      <input id="input_secondname" class="input_account"
        value="<? echo $_SESSION['user_data']['secondname'] ?>" type="text">
      <div class="decoration_line"></div>
      <p class="text_account">Почта</p>
      <input id="input_email" class="input_account"
        value="<? echo $_SESSION['user_data']['email'] ?>" type="text">
    </div>
  </div>
</div>

```

Рисунок 41 – Отображение данных пользователя

Второй блок работает аналогично, только данные теперь сначала внести, так как они ещё не записаны в базе данных.

Блок с изменением пароля не отличается функционалом от остальных. Для реализации данного блока использовался AJAX, а также метод `password_verify` который сверяет пароли (рис. 42, 43).

```

<?
require $_SERVER['DOCUMENT_ROOT'] . '/includes/connect_db.php';

$errors = array();

if (!password_verify($_POST['this_password'], $_SESSION['user_data']['password'])) {
    array_push($errors, 'Неверный пароль');
}
if ($_POST['new_password'] != $_POST['confirm_new_password']){
    array_push($errors, 'Пароли не совпадают');
}

if (!empty($errors)) {
    echo $errors[0] . $_POST['errors'];
}

if (empty($errors)) {
    $user = R::load('users', $_SESSION['user_data']['id']);
    $user->password = password_hash($_POST['new_password'], PASSWORD_DEFAULT);
    R::store($user);
    $user = $user->export();
    $_SESSION['user_data']=$user;
    array_push($errors, 'Данные обновлены!');
    echo $errors[0] . $_POST['errors'];
}

```

Рисунок 42 – Изменение пароля

```

$('body').on('click', '.save_password_button', function () {
$.ajax({
url: '/ajax/save_new_password.php',          /* Куда пойдет запрос */
method: 'post',                             /* Метод передачи (post или get) */
dataType: 'html',                           /* Тип данных в ответе (xml, json, script, html). */
data: {
this_password: $('#this_password').val(),
new_password: $('#new_password').val(),
confrim_new_password: $('#confrim_new_password').val()
}, /* Параметры передаваемые в запросе. */
success: function (data) { /* функция которая будет выполнена после успешного запроса. */
$('#errors_edit_profile').html(data); /* В переменной data содержится ответ от index.php. */
}
});
})

```

Рисунок 43 – AJAX запрос изменения пароля

Для реализации функционала вывода сообщений об ошибках и применения изменения пароля без обновления страницы был использован AJAX-запрос. Этот подход позволяет значительно улучшить пользовательский опыт, делая взаимодействие с веб-приложением более плавным и интерактивным.

Когда пользователь отправляет форму изменения пароля, AJAX-запрос отправляет данные на сервер асинхронно, без необходимости перезагрузки страницы. Сервер обрабатывает запрос, выполняет необходимые проверки и возвращает результат обратно в браузер. Если возникают ошибки, они передаются обратно и отображаются пользователю, не прерывая его взаимодействие с приложением. (рис. 44).

```

<?php foreach ($posts as $info_post): ?>
<? if ($SESSION['user_data']['email'] == $info_post['email_post']): ?>
<div class="col-12">
<div class="post_block">
<div class="row text-center">
<div class="col-1"><?php echo $info_post['id'] ?></div>
<div class="col-3"><?php echo $info_post['reason_post'] ?></div>
<div class="col-3">
<div data-id-description="<?php echo $info_post['id'] ?>" class="description_post_scroll" data-bs-toggle="modal"
data-bs-target="#description_post_modal">
<?php echo $info_post['description_post'] ?>
</div>
</div>
<div class="col-3"><?php echo $info_post['email_post'] ?></div>
<div class="col-2"><?php echo $info_post['date_post'] ?></div>
</div>
</div>
</div>
<? endif; ?>
<?php endforeach; ?>

```

Рисунок 44 – Вывод обращений пользователя

Модальное окно создания обращения создаёт AJAX запрос для вывода сообщений об ошибке без обновления страницы (рис.45).

```
$('#modal_send_message').on('click', '.modal_button_send', function () {
    $.ajax({
        url: '/ajax/test.php',          /* Куда пойдёт запрос */
        method: 'post',                /* Метод передачи (post или get) */
        dataType: 'html',              /* Тип данных в ответе (xml, json, script, html). */
        data: {
            description_message: $('#modal_textarea_description').val(),
            reason_message: $('#modal_input_reason').val(),
            email_message: $('#modal_email_post').val()
        }, /* Параметры передаваемые в запросе. */
        success: function (data) { /* функция которая будет выполнена после успешного запроса. */
            $('#errors_line').html(data); /* В переменной data содержится ответ от index.php. */
        }
    });
});
});
```

Рисунок 45 – AJAX запрос модального окна создания обращения

Для отправки обращения введенные данные должны пройти валидацию. Для поля «Причина» должны использоваться только символы кириллицы, длина должна быть больше 6, но меньше 50 символов. Для описания аналогичные критерии, только длина должна быть больше 20 символов, но меньше 1000 символов (рис. 46).

```
if (!preg_match('/[А-Яа-я]/', $_POST['reason_message'])) {
    array_push($errors, 'Введите корректную причину');
}
if (mb_strlen($_POST['reason_message'], 'UTF-8') < 6) {
    array_push($errors, 'Длина причины не может быть меньше 6 символов');
}
if (mb_strlen($_POST['reason_message'], 'UTF-8') > 50) {
    array_push($errors, 'Длина причины не может превышать 50 символов');
}
if (!preg_match('/[А-Яа-я]/', $_POST['description_message'])) {
    array_push($errors, 'Введите корректную причину');
}
if (mb_strlen($_POST['description_message'], 'UTF-8') < 20) {
    array_push($errors, 'Длина описания не может быть меньше 20 символов');
}
if (mb_strlen($_POST['description_message'], 'UTF-8') > 1000) {
    array_push($errors, 'Длина описания не может превышать 1000 символов');
}
if ($_POST['email_message'] != $_SESSION['user_data']['email']) {
    array_push($errors, 'Email не совпадают!');
}
```

Рисунок 46 – Валидация обращения

Если же ошибок не было найдено, то обращение записывается в базу данных, присваивается статус и создаются новые атрибуты для обработки (рис. 47).

```
if (empty($errors)) {
    array_push($errors, 'Заявка принята!');
    echo $errors[0] . $_POST['errors'];
    $post = R::dispense('posts');
    $post->date_post = date('Y-m-d');
    $post->email_post = $_POST['email_message'];
    $post->description_post = $_POST['description_message'];
    $post->reason_post = $_POST['reason_message'];
    $post->status_post = 'Новое';
    $post->number_office = $_POST[''];
    $post->date_post_write = $_POST[''];
    $post->explanation = $_POST[''];
    R::store($post);
}
```

Рисунок 47 – Запись данных в базу данных

После создания, обращение отобразится на панели администратора. Для вывода всех обращений был использован цикл foreach() (рис. 48).

```
<?php foreach ($posts as $info_post): ?>
    <? if ($_SESSION['user_data']['key_id'] == true): ?>
        <div class="col-12">
            <div id="<?php echo $info_post['status_post'] ?>" class="management_message">
                <div class="row text-center align-items-center">
                    <div class="col-1">
                        <p><?php echo $info_post['id'] ?></p>
                    </div>
                    <div class="col-4">
                        <p><?php echo $info_post['email_post'] ?></p>
                    </div>
                    <div class="col-2"><input class="management_date" type="date"
                        value="<?php echo $info_post['date_post'] ?>" disabled></div>
                    <div class="col-2">
                        <p><?php echo $info_post['status_post'] ?></p>
                    </div>
                    <div class="col-2 text-center">
                        <div data-id-message="<?php echo $info_post['id'] ?>"
                            class="check_button_management" data-bs-toggle="modal"
                            data-bs-target="#modal_check_message">
                            Просмотр
                        </div>
                    </div>
                </div>
            </div>
        </div>
    <? endif; ?>
<?php endforeach; ?>
```

Рисунок 48 – Вывод всех обращений на панели администратора

Для сортировки обращений от наибольшего к наименьшему был использован запрос, представленный на рисунке 49.

```
$posts = R::findAll('posts', "ORDER BY id DESC");
```

Рисунок 49 – Сортировка обращений

Сортировка была сделана для того, чтобы изначально шли новые обращения. Модальное окно редактирования обращений использует AJAX запросы для отправки ответа на обращение или его закрытия (рис. 50, 51).

```
$('body').on('click', '.modal_button_send', function () {
    $.ajax({
        url: '/ajax/send_answer.php',
        method: 'post',
        dataType: 'html',
        data: {
            select_number_post: $('#select_number_post').val(),
            date_post_write: $('#date_post_write').val(),
            explanation: $('#explanation').val(),
            id_post: $id_post
        },
        success: function (data) {
            $('#errors_line').html(data);
        }
    });
});
```

Рисунок 50 – AJAX запрос для ответа на обращение

```
$('body').on('click', '.modal_button_dismiss', function () {
    $.ajax({
        url: '/ajax/send_answer.php',
        method: 'post',
        dataType: 'html',
        data: {
            id_post: $id_post
        },
        success: function (data) {
            $('#errors_line').html(data);
        }
    });
});
```

Рисунок 51 – AJAX запрос для отклонения обращения

Вывод модального окна осуществляет тоже посредством использования AJAX запроса (рис. 52).


```

$( 'body' ).on( 'click', '.check_button_management', function () {
    $id_post = $( this ).attr( 'data-id-message' );
    $.ajax( {
        url: '/ajax/post.php',
        method: 'post',
        dataType: 'html',
        data: { id_post: $( this ).attr( 'data-id-message' ) },
        success: function ( data ) {
            $( '.modal_dialog_ajax' ).html( data );
        }
    } );
} );
}

```

Рисунок 52 – Открытие модального окна

Вывод модального окна с помощью AJAX запроса сделан для того, чтобы выводить данные относящиеся к данному обращению и вывода ошибок без обновления страницы (рис. 53).

```

require $ _SERVER[ 'DOCUMENT_ROOT' ] . '/includes/connect_db.php';

$post = R::load( 'posts', $ _POST[ 'id_post' ] );
$offices = R::findAll( 'offices' );

>>

<div class="modal-content">
  <div class="modal-header">
    <h1 class="modal-title fs-5" id="exampleModalLabel">Обращение № <? = $post[ 'id' ] ?></h1>
    <button type="button" class="btn-close" data-bs-dismiss="modal" aria-label="Close"></button>
  </div>
  <div class="modal-body modal-body-padding">
    <div class="row">
      <div class="col-12">
        <p class="text_account">Причина обращения</p>
        <input class="modal_input" type="text" value="<? = $post[ 'reason_post' ] ?>" disabled>
        <div class="decoration_line"></div>
        <p class="text_account">Описание проблемы</p>
        <textarea class="modal_textarea" cols="30" rows="5" disabled><? = $post[ 'description_post' ] ?></textarea>
        <p class="text_account">Адрес электронной почты</p>
        <input class="modal_input" type="text" value="<? = $post[ 'email_post' ] ?>" disabled>
        <div class="decoration_line"></div>
        <p class="text_account">В какой кабинет направить?</p>
        <select class="select_modal" name="cabinet" id="select_number_post">
          <option value="">нет</option>
          <?php foreach ( $offices as $info office ): ?>
            <? if ( $ _SESSION[ 'user_data' ] ): ?>
              <option value="<?php echo $info_office[ 'number_office' ] ?>"><?php echo $info_office[ 'number_office' ] ?></option>
            <? endif; ?>
          <?php endforeach; ?>
        </select>
        <p class="text_account">На какое число записать?</p>
        <input class="input_modal" type="date" id="date_post_write">
        <p class="text_account">Пояснение для заявителя</p>
        <textarea class="modal_textarea" name="" id="explanation" cols="30" rows="5"></textarea>
        <div class="errors_line text-center"></div>
      </div>
    </div>
  </div>
  <div class="modal-footer">
    <button type="button" class="modal_button_send">Записать</button>
    <button type="button" class="modal_button_dismiss" data-bs-dismiss="modal">Отклонить</button>
    <button type="button" class="modal_button_close" data-bs-dismiss="modal">Закрыть</button>
  </div>
</div>

```

Рисунок 53 – Вывод модального окна обращений

Для отправки ответа вводимые данные должны пройти валидацию. В данном случае используется «Исключающее ИЛИ» для полей «Кабинет» и «Дата записи». Если «Кабинет» не пустой, то «Дата записи» тоже должна быть не пустой и наоборот (рис. 54).

```
if((empty($_POST['select_number_post']) && !empty($_POST['date_post_write']) || (!empty($_POST['select_number_post']) && empty($_POST['date_post_write'])))
{
    array_push($errors, 'Выберите кабинет и число записи');
}
```

Рисунок 54 – Валидация для «Кабинет» и «Дата»

Если не ничего не вводить, то обращение будет рассмотрено и пользователь получит сообщение на почту «Ваше обращение рассмотрено», если ввести «Кабинет» и «Дата записи», то пользователь получить сообщение на почту «Ваше обращение находится на рассмотрении, явитесь [дата записи] в [номер кабинета]» (рис. 55)

```
if(empty($errors))
{
    if(empty($_POST['select_number_post']) && empty($_POST['date_post_write']) && empty($_POST['explanation'])){
        $post->status_post = 'Закрыт';
        R::store($post);
        array_push($errors, 'Ответ отправлен!');
        echo $errors[0] . $_POST['errors'];
    }
    else{
        $post->number_office = $_POST['select_number_post'];
        $post->date_post_write = $_POST['date_post_write'];
        $post->explanation = $_POST['explanation'];
        $post->status_post = 'Ожидает';
        R::store($post);
        array_push($errors, 'Запись создана!');
        echo $errors[0] . $_POST['errors'];
    }
}
```

Рисунок 55 – Отправка ответа

Для управления кабинетами было создано отдельное модальное окно позволяющее создавать и удалять кабинеты. Модальное окно также действует через AJAX запросы для вывода ошибок без обновления страницы.

Вводимые данные в данном модальном окне проходят валидацию, как и в предыдущих случаях. Для поля «Номер кабинета» требуется, чтобы вводимые данные содержали только цифры, и длина не превышала 3 символа (рис. 56).

```

if (preg_match('/^[0-9]/', $_POST['number_office']) || empty($_POST['number_office'])) {
    array_push($errors, 'Введите корректный номер кабинета');
}
if (mb_strlen($_POST['number_office'], 'UTF-8') > 3) {
    array_push($errors, 'Длина номера не может превышать 3 символа');
}

```

Рисунок 56 – Валидация поля «Номер кабинета»

Для поля «ФИО заведующего» требуется, чтобы вводимые данные содержали только символы кириллицы, а также длина была больше 2 символов, но не превышала 40 символов (рис. 57).

```

if (!preg_match('/[А-Яа-я]/', $_POST['supervisor_office'])) {
    array_push($errors, 'Введите корректное ФИО');
}
if (mb_strlen($_POST['supervisor_office'], 'UTF-8') < 2) {
    array_push($errors, 'Длина ФИО не может быть меньше 2 символов');
}
if (mb_strlen($_POST['supervisor_office'], 'UTF-8') > 40) {
    array_push($errors, 'Длина ФИО не может превышать 40 символов');
}

```

Рисунок 57 – Валидация поля «ФИО заведующего»

Для поля «Вид деятельности» необходимо, чтобы вводимые данные содержали только символы кириллицы и были больше 6 символов, но не превышали 100 символов (рис. 58).

```

if (!preg_match('/[А-Яа-я]/', $_POST['type_of_activity'])) {
    array_push($errors, 'Введите корректный вид деятельности');
}
if (mb_strlen($_POST['type_of_activity'], 'UTF-8') > 100) {
    array_push($errors, 'Длина описания не может превышать 100 символов');
}
if (mb_strlen($_POST['type_of_activity'], 'UTF-8') < 6) {
    array_push($errors, 'Длина описания не может быть меньше 6 символов');
}

```

Рисунок 58 – Валидация поля «Вид деятельности»

Если ошибок не было найдено, кабинет сохраняется в базу данных (рис. 59).

```

if (empty($errors)) {

    array_push($errors, 'Кабинет успешно создан!');
    echo $errors[0] . $_POST['errors'];
    $office = R::dispense('offices');
    $office->number_office = $_POST['number_office'];
    $office->supervisor_office = $_POST['supervisor_office'];
    $office->type_of_activity = $_POST['type_of_activity'];
    R::store($office);

}

```

Рисунок 59 - Сохранение данных в базу данных

Для удаления кабинета реализован AJAX запрос, который выполняет удаление кабинета из базы данных по его идентификатору (рис. 60).

```

$office = R::load('offices', $_POST['id_office']);

R::trash($office);

echo 'Успешно!' . $_POST['errors'];

```

Рисунок 60 – Удаление кабинета

Для вывода всех кабинетов в меню управления использован цикл foreach (рис. 61).

```

<?php foreach ($offices as $info_office): ?>
<? if ($_SESSION['user_data']['key_id'] == true): ?>
<div class="col-12">
<div class="row text-center">
<div class="col-2"><div class="text_scroll"><?php echo $info_office['number_office'] ?></div>
</div>
<div class="col-4"><div class="text_scroll"><?php echo $info_office['supervisor_office'] ?></div>
</div>
<div class="col-5"><div class="text_scroll"><?php echo $info_office['type_of_activity'] ?></div>
</div>
<div class="col-1">
<div data-id-office="<?php echo $info_office['id'] ?>" class="delete_office">
<svg
xmlns="http://www.w3.org/2000/svg"
fill="currentColor" class="bi bi-trash" viewBox="0 0 16 16">
<path
d="M5.5 5.5A.5.5 0 0 1 6 6v6a.5.5 0 0 1 1 1v6a.5.5 0 0 1 .5.5m2.5 0a.5.5 0 0 1 .5v6a.5.5 0 0 1 1 1v6a.5.5 0 0 1 .5.5m3 .5a.5.5 0 0 1
<path
d="M14.5 3a1 1 0 0 1-1 1H13v9a2 2 0 0 1-2 2H5a2 2 0 0 1-2-2V4h-.5a1 1 0 0 1-1-1V2a1 1 0 0 1 1-1H6a1 1 0 0 1 1-1h2a1 1 0 0 1 1-1h3.5a1 1 0
</svg>
</div>
</div>
</div>
<? endif; ?>
<?php endforeach; ?>

```

Рисунок 61 – Вывод всех кабинетов

2.5 Проектирование интерфейса

Дизайн сайта администрации должен быть функциональным, интуитивно понятным и привлекательным, чтобы обеспечить удобство использования для всех категорий граждан.

Шапка (Header) должна содержать логотип администрации, главное меню навигации и кнопку поиска.

Основной контент (Main Content) включает слайдер с актуальными новостями и объявлениями, блоки с основными разделами (Обращения, Бюджет, Новости, Контакты), интерактивный календарь событий и последние новости, и обновления.

Боковая панель (Sidebar) предоставляет быструю навигацию по разделам, ссылки на социальные сети и виджет обратной связи.

Подвал (Footer) содержит контактную информацию, карту сайта, политику конфиденциальности и ссылки на правовые документы.

Наиболее удобным сервисом для создания интерфейса сайта является Figma.

Создание интерфейса сайта администрации в Figma начинается с исследования и планирования. Важно определить целевую аудиторию, её потребности, а также изучить лучшие практики аналогичных сайтов. Сначала создаётся схема сайта (site map), чтобы определить структуру и основные разделы, а затем ключевые функции и требования к интерфейсу.

Следующим этапом является создание вайрфреймов, которые представляют собой скелетные наброски страниц. Сначала определяется основной макет страницы с использованием простых прямоугольников и линий для обозначения различных блоков: шапки, основного содержания, боковой панели и подвала. Затем размещаются ключевые элементы, такие как логотип, меню навигации, кнопка поиска, блоки контента и формы (рис. 62).

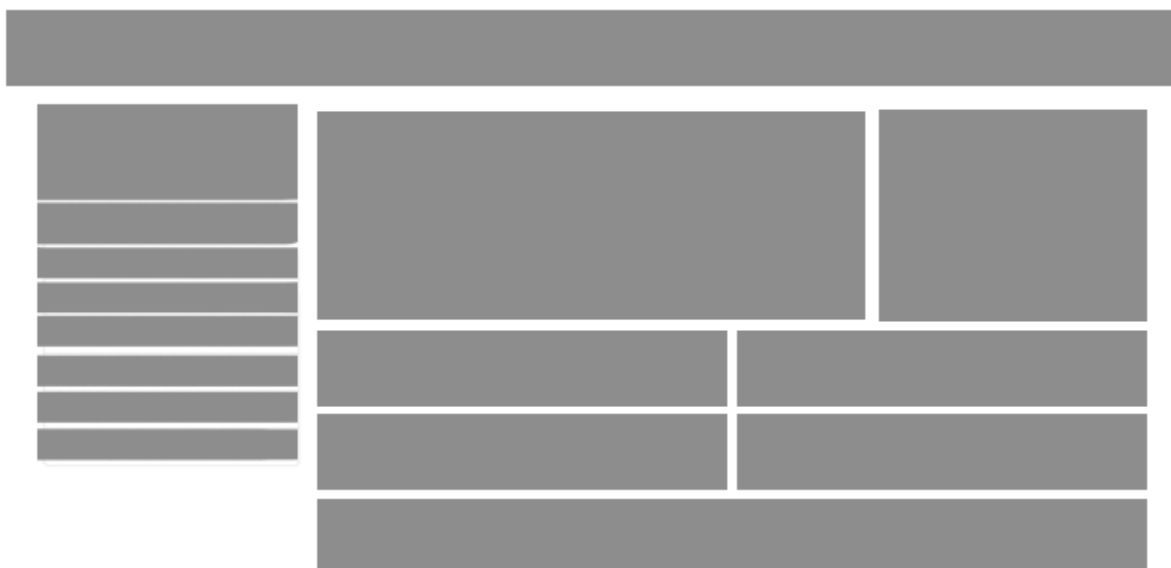


Рисунок 62 – Шаблон структуры сайта

На этапе создания стиля и визуальных элементов определяется визуальный стиль сайта. Выбирается цветовая палитра, которая соответствует бренду администрации, шрифты для заголовков и основного текста, а также иконки и изображения, которые будут использоваться на сайте. Важно, чтобы все эти элементы были легко читаемыми и соответствовали общей эстетике сайта (рис. 63).



Рисунок 63 – Цветовая палитра сайта

Проектирование в Figma включает создание компонентов, которые будут использоваться повторно, таких как кнопки, формы и карточки новостей. Это обеспечивает консистентность и удобство внесения изменений. Далее, на основе вайрфреймов макетируются страницы, добавляются визуальные эле-

менты, цвета и шрифты. Применение автолейаутов помогает адаптировать дизайн под различные устройства, обеспечивая адаптивность элементов (рис. 64).

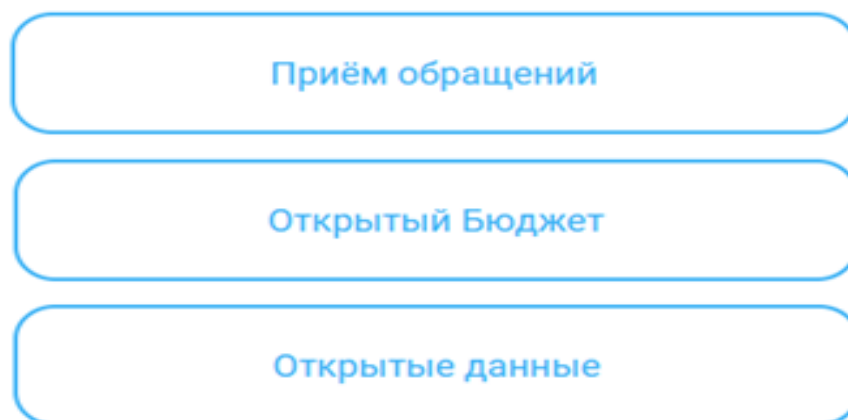


Рисунок 64 – Пример использования повторных компонентов Интерфейс взаимодействия и прототипирование включают создание интерактивных прототипов с кликабельными элементами, такими как ссылки и кнопки. Важно протестировать пользовательский путь, чтобы убедиться в интуитивности и логичности навигации.

Конечный дизайн сайта изображен на рисунке 65.

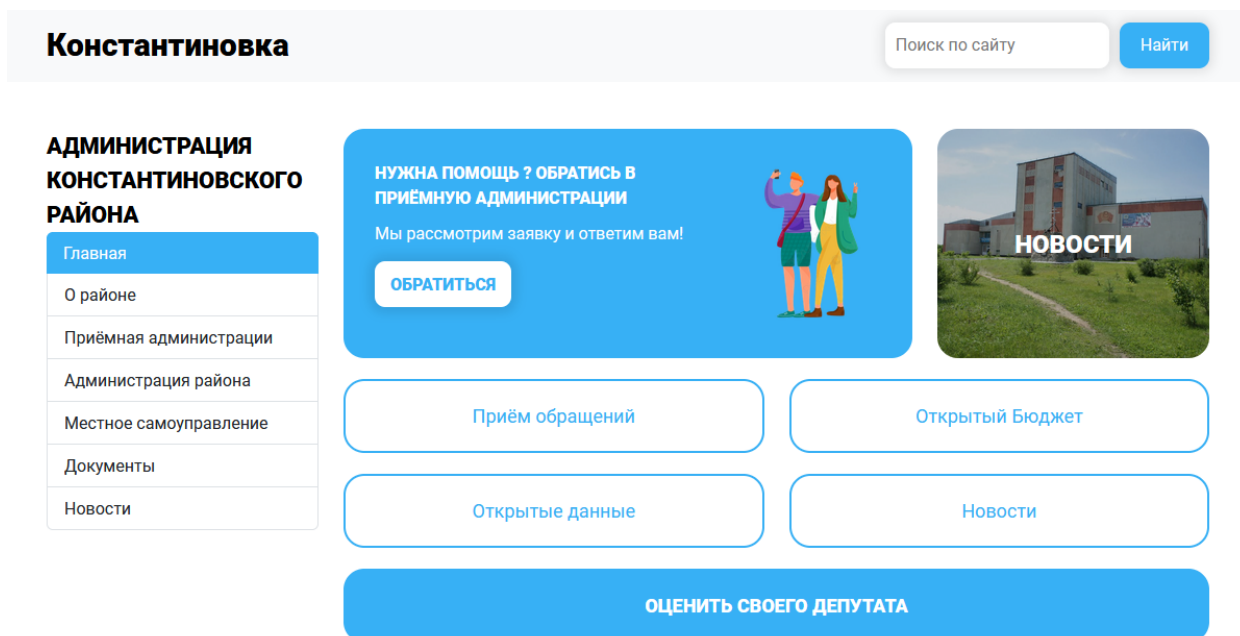
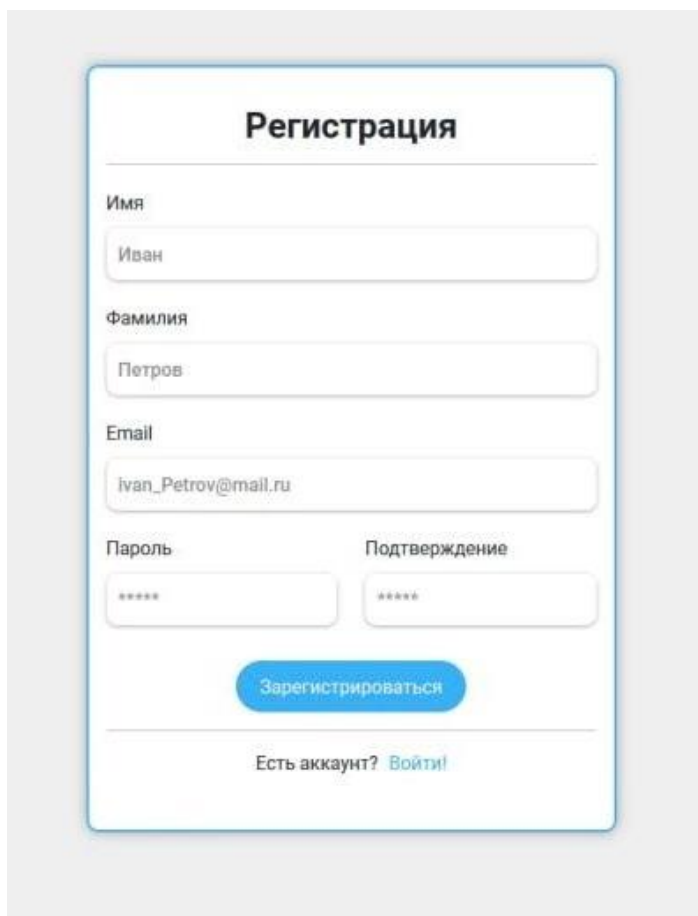


Рисунок 65 – Итоговый интерфейс сайта

2.6 Тестирование программного обеспечения

Для того чтобы войти на сайт, необходимо пройти регистрацию, которая включает несколько шагов. Пользователям потребуется ввести свои имя и фамилию, что помогает администрации идентифицировать заявителя и обеспечивать персонализированный подход к каждому обращению.

Для регистрации необходимо указать адрес электронной почты, номер телефона, ввести пароль и подтвердить его. В данном случае сперва производится проверка введённого адреса электронной почты и номера телефона, если данные значения были найдены в базе данных, то выведется соответствующее сообщение и предложение войти в аккаунт. Затем пользователю необходимо создать пароль, который будет использоваться для защиты его аккаунта. При создании пароля рекомендуется выбирать комбинацию, включающую буквы, цифры и специальные символы для обеспечения максимальной безопасности личных данных (рис. 66).



The image shows a registration form with the following fields and elements:

- Имя** (Name): Input field containing "Иван".
- Фамилия** (Surname): Input field containing "Петров".
- Email**: Input field containing "ivan_petrov@mail.ru".
- Пароль** (Password): Input field with masked characters "*****".
- Подтверждение** (Confirmation): Input field with masked characters "*****".
- Зарегистрироваться** (Register): A blue button.
- Есть аккаунт? Войти!** (Have an account? Log in!): A link at the bottom of the form.

Рисунок 66 – Окно регистрации

После регистрации для доступа к сайту пользователю необходимо авторизоваться, введя свои логин и пароль. Логин может служить адрес электронной почты, указанный при регистрации, что упрощает процесс входа и уменьшает вероятность забывания логина. Введенный пароль, созданный пользователем во время регистрации, обеспечивает безопасность его учетной записи (рис. 67).

Рисунок 67 – Страница авторизации

После ввода всех необходимых данных и завершения регистрации пользователь получает доступ ко всем функциям сайта, включая подачу обращений, участие в голосованиях и прочее (рис. 68).

Рисунок 68 – Главная страница

На странице настройки аккаунта, на вкладке «Основные» имеется возможность изменить логин, адрес электронной почты, номер телефона и загрузить фотографию, при загрузке изображения, загрузка на сайт происходит автоматически, без подтверждения нажатием кнопки «Сохранить». На вкладке «Информация о себе» можно изменить имя, фамилию или рассказать о себе. Изменить пароль соответственно. При нажатии кнопки «Сохранить» отправляется запрос в базу данных с изменением данных для текущего пользователя (рис. 69).

Константиновка

Поиск по сайту

Найти

Основные Информация о себе Обращения Изменить пароль

Логин

Почта

Номер телефона

Загрузить новое фото

Только PNG, JPG, GIF

Сохранить Отменить

Рисунок 69 – Страница настройки аккаунта

На вкладке «Обращения» пользователь имеет доступ к полному списку своих текущих обращений, где можно видеть их статусы, даты подачи и последние обновления. Это позволяет легко отслеживать ход рассмотрения каждого обращения и быть в курсе всех изменений. Кроме того, на этой вкладке пользователь имеет возможность создавать новые обращения. При создании нового обращения необходимо указать тему и причину обращения, подробно описать возникшую проблему или вопрос, а также ввести контактные данные для обратной связи, такие как адрес электронной почты. Это

обеспечивает администрацию всей необходимой информацией для оперативного и точного ответа (рис. 70).

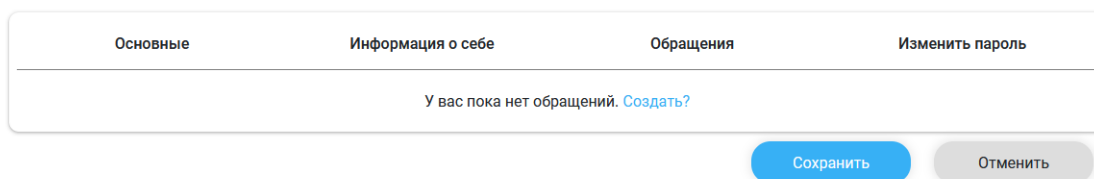


Рисунок 70 – Окно просмотра обращений

В окне создания обращения пользователю нужно указать несколько важных деталей. Прежде всего, следует выбрать или ввести причину обращения, что поможет администрации быстро понять суть запроса или проблемы. Затем необходимо подробно описать проблему, предоставив как можно больше информации для точного и эффективного рассмотрения.

После этого пользователь должен указать свой адрес электронной почты. Это необходимо для обеспечения обратной связи, позволяя администрации отправлять уведомления о статусе обращения, запросы дополнительной информации или ответы на запросы. Электронная почта также служит средством для подтверждения получения обращения и дальнейшего взаимодействия (рис. 71).

The image shows a modal window titled "Создать обращение" with a close button (X) in the top right corner. It contains three input fields: "Введите причину обращения" with the value "Засор водоканала"; "Опишите проблему" with the value "Засорился водоканал, на улице Ленина 12. Мешает уходу воды, в проходимом месте"; and "Адрес электронной почты" with the value "sdgfsdg@mail.ru". Below the fields, a red message says "Заявка принята!". At the bottom, there are two buttons: "Отправить" (highlighted in blue) and "Закреть" (greyed out).

Рисунок 71 – Форма обращения

После создания обращения оно должно пройти проверку модератором. Когда пользователь с определёнными правами авторизуется на сайте, у него появляется возможность зайти на страницу управления обращениями. На данной странице отображаются все обращения, независимо от их статуса и периода времени подачи. Для удобства работы с обращениями предусмотрено боковое меню быстрой сортировки по статусу обращения, что позволяет мгновенно фильтровать обращения по их текущему состоянию, например, новые, в процессе рассмотрения или завершённые (рис. 72).

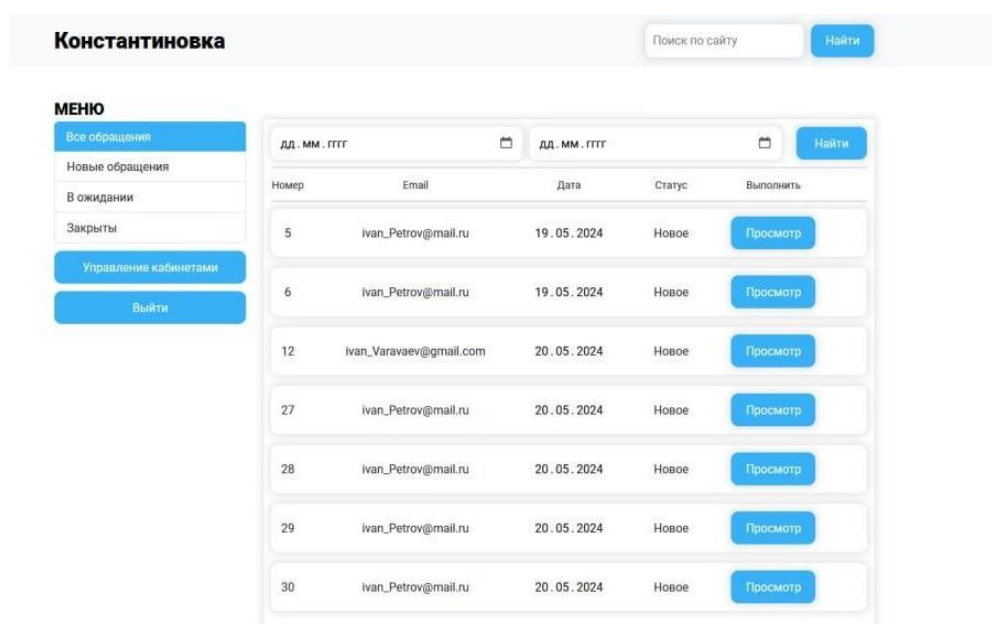


Рисунок 72 – Меню обращений

Дополнительно, для более точного поиска и управления обращениями, на странице имеется возможность ввода необходимого периода времени, за который необходимо вывести все обращения. Пользователь может задать конкретные даты начала и окончания периода, что позволяет сфокусироваться на обращениях, поданных в интересующий временной интервал. Для вывода обращений за выбранный промежуток времени система отправляет запрос в базу данных, сравнивая указанный диапазон дат с датой каждого обращения. Если дата обращения попадает в заданный диапазон, то оно будет отображено на странице. Такой подход обеспечивает точную и эффективную

фильтрацию, позволяя пользователю легко находить и обрабатывать необходимые обращения.

Модератор, нажав на кнопку «Управление кабинетами», получает доступ к функционалу редактирования данных о кабинетах. Это включает возможность изменять номер кабинета, фамилию, имя и отчество (ФИО) заведующего, а также вид деятельности кабинета.

При изменении номера кабинета модератор может обновить информацию для точного соответствия текущему распределению помещений, что важно для поддержания актуальной базы данных и облегчения навигации как для сотрудников, так и для посетителей. Изменение ФИО заведующего позволяет своевременно обновлять данные в случае кадровых изменений. Это важно для поддержания точности информации о руководителях каждого кабинета, что способствует эффективному взаимодействию и коммуникации внутри организации. Кроме того, модератор может обновить вид деятельности кабинета, что позволяет актуализировать информацию о том, какие услуги или функции выполняются в каждом конкретном кабинете (рис. 70).

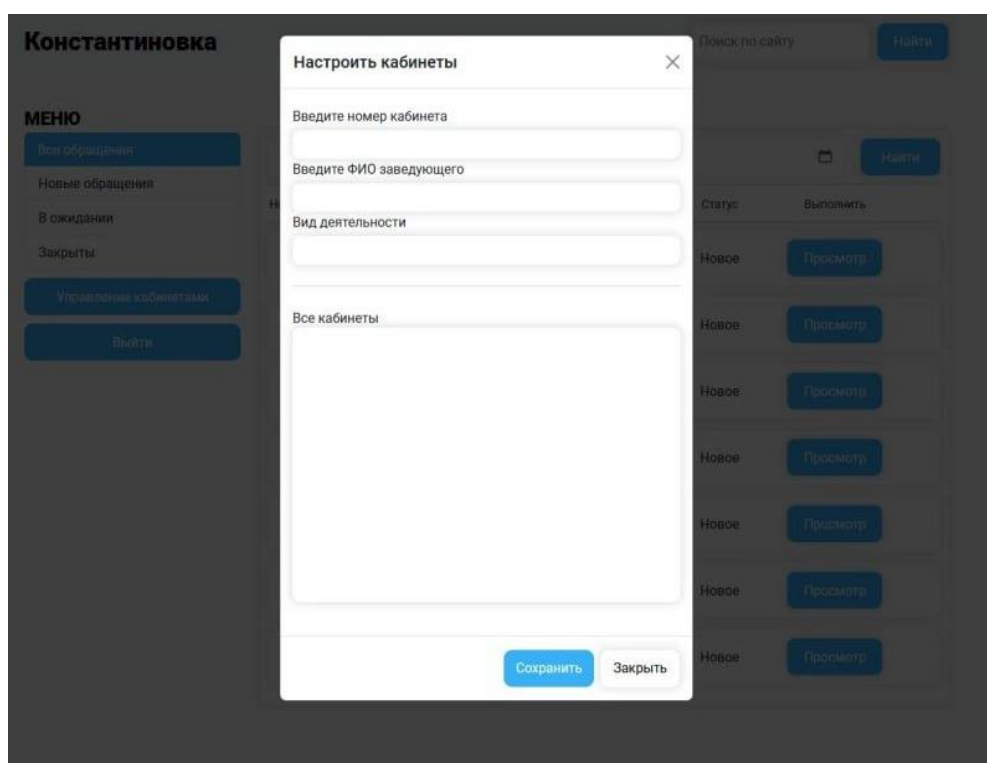


Рисунок 73 – Настройка кабинета

В окне просмотра обращения за оператором стоит задача качественного анализа обращения. Оператор также должен указать конкретную дату и кабинет, в который пользователю необходимо явиться для решения его вопроса (рис. 74).

Обращение №

Причина обращения
Засор водоканала

Описание проблемы
Засорился водоканал, на улице Ленина 12, Мешает уходу воды, в проходимом месте

Адрес электронной почты
sdgfsdg@mail.ru

В какой кабинет направить?
101

На какое число записать?
дд . мм . гггг

Записать Отклонить Закреть

Рисунок 74 – Окно дополнительной информации об обращении

После определения всех необходимых деталей, обращение переводится в статус «Ожидание». В этот момент информация о назначенной дате и кабинете сохраняется в базе данных, обеспечивая систематизацию и возможность последующего отслеживания. Пользователь получает уведомление с указанными данными, что позволяет ему подготовиться и своевременно посетить назначенное место.

Когда назначенная дата истекает, данные о записи автоматически удаляются из базы данных, и статус обращения изменяется на «Закрывает». Это автоматическое обновление статуса помогает поддерживать базу данных в ак-

туальном состоянии, устраняя устаревшую информацию и обеспечивая точное отражение текущего состояния всех обращений.

Если оператор решает, что обращение не требует дальнейших действий, он может нажать на кнопку «Отклонить». В этом случае обращение сразу же переводится в статус «Закрыт», и пользователь получает соответствующее уведомление.

При нажатии на кнопку «Закрыть» модальное окно просто закрывается без внесения каких-либо изменений в обращение. Это позволяет оператору вернуться к предыдущему экрану или другим задачам без воздействия на текущее обращение, сохраняя его статус и все введенные данные неизменными.

3 БЕЗОПАСНОСТЬ И ЭКОЛОГИЧНОСТЬ

3.1 Безопасность

Безопасность жизнедеятельности (БЖД) представляет собой широкий дисциплинарный раздел, охватывающий как теоретические, так и практические аспекты защиты человека от опасных и вредных факторов, которые могут возникать в различных сферах его деятельности. Ее основной задачей является обеспечение сохранения безопасности и здоровья людей в их окружающей среде путем защиты от воздействия как антропогенных (связанных с деятельностью человека), так и природных факторов.

В контексте работы с компьютерной техникой, знание правил техники безопасности имеет большое значение для операторов. Эти знания позволяют:

Предотвратить несчастные случаи:

– правильное обращение с электроприборами и соблюдение норм безопасности при работе с электрическими устройствами помогает избежать поражения электрическим током;

– соблюдение правил пожарной безопасности при использовании компьютерной техники, таких как правильное размещение проводов и использование сертифицированных розеток и удлинителей, предотвращает возникновение пожаров.

Уменьшить риски воздействия вредных факторов:

– правильная организация рабочего пространства, включая размещение компьютера, монитора, клавиатуры и стула, помогает уменьшить физическое напряжение и предотвратить мышечно-скелетные расстройства;

– обеспечение надлежащего освещения и вентиляции в помещении снижает риск утомления глаз и улучшает общее самочувствие.

Предупредить развитие профессиональных заболеваний:

– регулярные перерывы и правильное размещение монитора помогают снизить напряжение глаз и предотвратить синдром компьютерного зрения;

– использование эргономичных клавиатур и мышей, а также правильное положение рук при работе с компьютером, уменьшает риск развития карпального туннельного синдрома.

Безопасность жизнедеятельности в контексте работы с компьютерной техникой играет ключевую роль в предотвращении несчастных случаев, уменьшении рисков воздействия вредных факторов и предупреждении развития профессиональных заболеваний. Соблюдение требований к рабочим местам, оборудованным ПК, способствует созданию безопасных и здоровых условий труда, что в конечном итоге повышает производительность и общее благополучие работников.

3.1.1 Требования к помещению для работы с ПЭВМ

Создание сайтов предполагает продолжительную и интенсивную работу за компьютером, что может привести к различным вредным и опасным воздействиям на человека. Эти воздействия могут существенно повлиять на здоровье и общее самочувствие работников, если не соблюдать правила и стандарты безопасности.

В настоящее время действуют санитарные правила СП 2.2.3670-20, которые содержат требования к производственным зданиям, помещениям и сооружениям. Они включают следующие пункты:

– объем помещений на одного работника, предназначенных для постоянных рабочих мест, должен быть не менее 15 м³ при выполнении легких физических работ с категориями энергозатрат Ia–Ib. Это требование актуально независимо от характера выполняемой работы и установленных гигиенических нормативов энергозатрат. Достаточный объем воздуха в помещении обеспечивает нормальные условия для дыхания и способствует снижению утомляемости, что особенно важно для поддержания высокого уровня производительности и благополучия работников.

– площадь рабочих помещений на одного сотрудника, вне зависимости от характера выполняемой работы, должна составлять не менее 4,5 м². Это позволяет обеспечить каждому работнику достаточное личное пространство, необходимое для комфортного и безопасного выполнения своих обязанностей, снижая риск столкновений и травм, а также улучшая общую эргономику рабочего места;

– при размещении нескольких промышленных установок, генерирующих электромагнитное излучение (ЭМИ), в одном помещении, их размещение должно исключать возможность превышения гигиенических нормативов на рабочих местах в результате суммирования энергии излучения. Это необходимо для предотвращения негативного воздействия ЭМИ на здоровье сотрудников, которое может проявляться в виде головных болей, усталости, снижения концентрации и других симптомов;

– покрытия полов в местах, где возможно соприкосновение с агрессивными жидкостями (кислотами, щелочами, окислителями, растворителями, биологически активными веществами), а также с ртутью, должны быть устойчивыми к воздействию этих веществ и не должны их поглощать. Такие покрытия предотвращают разрушение пола и защищают сотрудников от потенциальных химических ожогов и отравлений, обеспечивая безопасность рабочей среды;

– для предотвращения попадания холодного воздуха в производственные помещения, входы в здания должны быть оборудованы системами, ограничивающими проникновение холодного воздуха снаружи. Такие системы включают воздушные завесы или тамбуры, которые помогают поддерживать комфортную температуру внутри помещений, что способствует сохранению здоровья сотрудников и поддержанию оптимальных условий для работы, особенно в холодное время года.

3.1.2 Требования к освещению на рабочих местах с ПЭВМ

Правильное освещение играет ключевую роль в создании комфортных условий для работы операторов с использованием электронно-вычислительных машин (ЭВМ). Оно напрямую влияет на здоровье, самочувствие и производительность труда. Для обеспечения оптимального освещения на рабочих местах с ЭВМ необходимо соблюдать следующие требования:

- естественный свет должен падать преимущественно слева, поэтому рабочие места следует располагать так, чтобы мониторы были установлены боком к оконным проемам;

- искусственное освещение должно быть равномерным и комбинированным в производственных помещениях. Освещенность экранов не должна превышать 300 люкс и создавать бликов. Освещенность рабочей зоны стола должна быть в пределах 300-500 люкс;

- яркость светящихся поверхностей: В поле зрения яркость не должна превышать 200 кд/м². Необходимо ограничивать прямую и отраженную блескость;

- яркость бликов на экране должна быть не более 40 кд/м², а от потолка – не более 200 кд/м²;

- в зоне углов освещения от 50 до 90 градусов яркость общего освещения не должна превышать 200 кд/м². Защитный угол светильников должен быть не менее 40 градусов;

- следует ограничивать неравномерность распределения яркости в рабочей зоне;

- люминесцентные светильники: общее освещение должно быть выполнено в виде параллельных линий, расположенных сбоку от рабочего места;

- коэффициент пульсации: Он не должен превышать 5%;

– необходимо своевременно заменять перегоревшие лампы и проводить регулярную чистку светильников и окон не реже двух раз в год.

Соблюдение данных требований к освещению, описанных также в СанПиН 1.2.3685-21, позволит создать благоприятные условия для работы с ЭВМ и снизить негативное влияние на здоровье операторов.

Правильное расположение источников света изображено на рисунке 75.

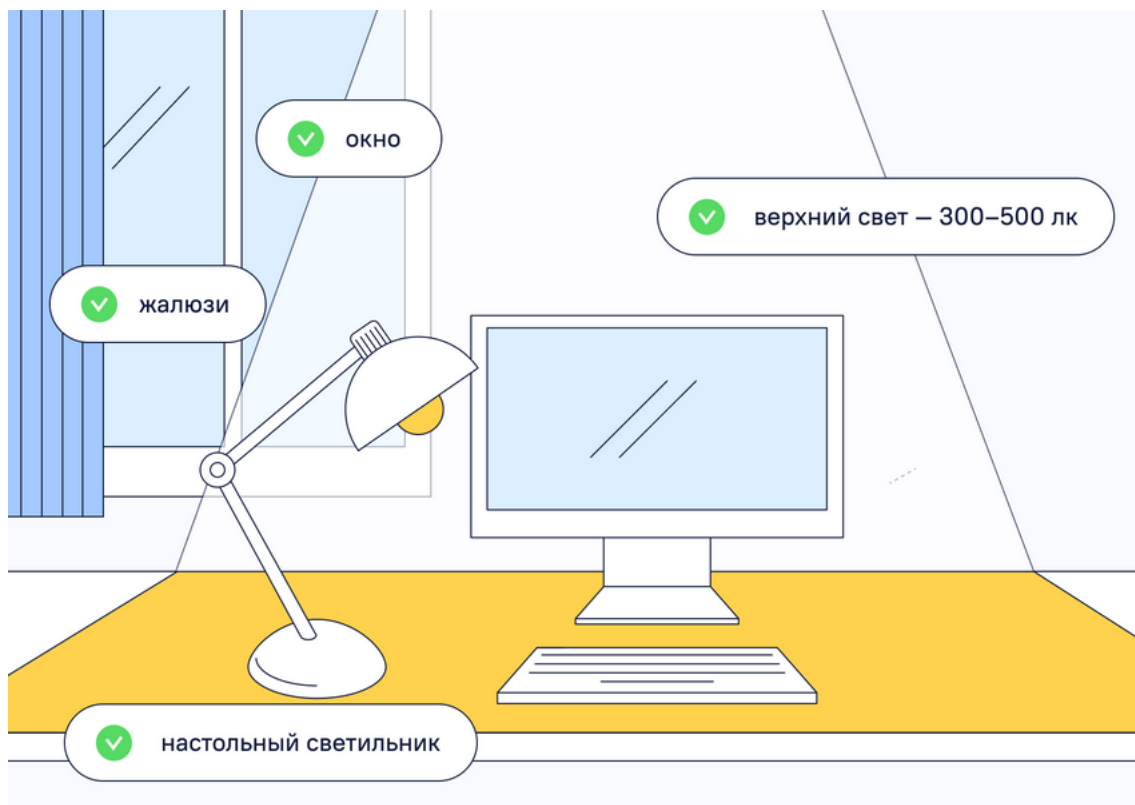


Рисунок 75 – Требования к освещению рабочих мест

3.1.3 Требования к организации рабочих мест с ПЭВМ

Исследования и практический опыт показывают, что правильная организация рабочего пространства имеет большое значение для работоспособности, комфорта и благополучия сотрудников. Это связано с несколькими факторами.

Эргономичная организация рабочего места позволяет сотрудникам сохранять правильную осанку, снижает риск физических перегрузок и травм. Правильно подобранная мебель и оборудование обеспечивают удобство и предотвращают перенапряжение мышц и суставов.

Оптимально организованное рабочее пространство способствует повышению производительности и эффективности работников. Хорошо организованная зона позволяет сосредоточиться на задачах, экономит время на поиск необходимых материалов и оборудования.

Комфортные условия труда, созданные оптимальным рабочим пространством, повышают удовлетворенность сотрудников и их общее благополучие. Удобная мебель, освещение, вентиляция и контроль температуры снижают уровень стресса и дискомфорта.

Правильная организация рабочих мест обеспечивает безопасность персонала: свободный доступ к аварийным выходам, предотвращение перегрузки электросетей, безопасное размещение оборудования и материалов.

Для сохранения здоровья и повышения эффективности операторов необходимо следовать следующим рекомендациям:

Санитарные правила и нормы СанПиН 2.2.2/2.4.1340-03 регламентируют требования к организации рабочих мест с компьютерами, включая площадь помещения на одно устройство. Согласно этим нормам, в помещении, где установлены компьютеры с ЖК-экранами, на каждое такое устройство должно приходиться не менее 4,5 м².

Это значит, что в помещении площадью 9 м² можно расположить только два рабочих места с мониторами, независимо от диагонали экрана. Таким образом, размер офиса определяется в большей степени числом мониторов, а не количеством сотрудников.

Соблюдение нормы 4,5 м² на каждое рабочее место позволяет создать комфортные и безопасные условия для работы. Это обеспечивает достаточное личное пространство для каждого сотрудника, что важно для предотвращения перенапряжения и стресса (рис. 76).

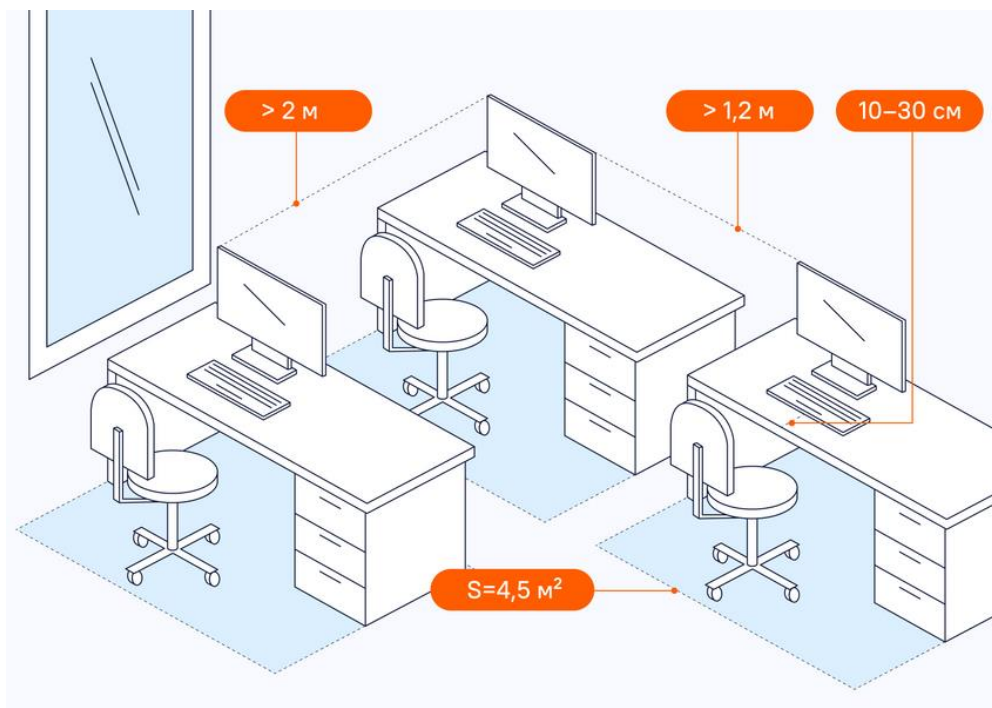


Рисунок 76 – Правило установления расстояния между мониторами

Кроме того, столы надо устанавливать так, чтобы расстояние между стоящими на них мониторами было не меньше 2 м спереди и не менее 1,2 м сбоку.

Если между столами установлены перегородки, их высота должна быть (1,5–2) м.

Естественный свет должен падать на рабочую поверхность слева. То есть размещать столы лучше друг за другом так, чтобы окно было по левую руку.

Рабочее кресло офисного сотрудника должно иметь минимум три регулировки:

- по высоте сиденья;
- по углу наклона спинки;
- по расстоянию спинки от переднего угла сиденья.

На кресле должно быть удобно сидеть, а его поверхность должна быть достаточно мягкой, нескользящей и легко очищаться от загрязнений.

Для сохранения здоровья и повышения эффективности операторов необходимо следовать следующим рекомендациям:

– рабочий стол должен быть регулируемым по высоте в диапазоне от 680 до 800 миллиметров или иметь стандартную высоту в 725 миллиметров. Рекомендуемые размеры: ширина от 800 до 1400 миллиметров, глубина от 800 до 1000 миллиметров;

– под столом должно быть свободное пространство для ног высотой не менее 600 миллиметров, шириной 500 миллиметров, глубиной на уровне колен не менее 450 миллиметров, а при вытянутых ногах - не менее 650 миллиметров;

– рабочий стул должен иметь сиденье шириной и глубиной не менее 400 миллиметров, регулируемую высоту от 400 до 550 миллиметров, наклон вперед до 15 градусов, назад до 5 градусов. Спинка должна быть высотой от 280 до 320 миллиметров, шириной 380 миллиметров, радиусом кривизны 400 миллиметров, с наклоном 30 градусов. Регулируемые подлокотники должны быть высотой от 200 до 260 миллиметров над сиденьем, расстоянием между ними от 350 до 500 миллиметров и длиной не менее 250 миллиметров;

– клавиатура должна быть расположена на расстоянии от края стола от 100 до 300 миллиметров;

– необходимо иметь подставку для ног размером шириной 300 миллиметров, глубиной 400 миллиметров, высотой 150 миллиметров, с наклоном до 20 градусов;

– расстояние между мониторами спереди должно быть не менее 2 метров, а сбоку - не менее 1,2 метров;

– экран монитора должен располагаться на расстоянии от глаз оператора от 600 до 700 миллиметров;

Соблюдение данных рекомендаций поможет создать оптимальные условия для работы операторов.

3.1.4 Требования к микроклимату рабочего места

Микроклиматические условия играют важнейшую роль при работе с электронно-вычислительными машинами (ЭВМ), оказывая прямое влияние на комфорт и продуктивность сотрудников. Поддержание оптимальных параметров микроклимата помогает уменьшить физическое и психологическое напряжение, связанное с работой на компьютере, и способствует общему благополучию.

Согласно санитарно-эпидемиологическим нормам (СанПиН 1.2.3685-21), установлены стандарты и показатели микроклимата на рабочих местах. Оптимальные микроклиматические условия определяются требованиями к комфортной температуре для человека, одетого в одежду с теплоизоляцией 1 кло в холодное время года и 0,7–0,8 кло в теплый период. Эти условия обеспечивают удовлетворительное тепловое состояние на протяжении рабочего дня, минимизируя нагрузку на механизмы терморегуляции, не влияя отрицательно на здоровье и способствуя высокой эффективности труда.

Допустимые микроклиматические условия также определяются требованиями к тепловому комфорту для человека, одетого в одежду с теплоизоляцией 1 кло в холодный период и 0,7–0,8 кло в теплый период на протяжении восьмичасовой рабочей смены. Эти условия не представляют угрозы для здоровья, но могут вызывать некоторое чувство дискомфорта или напряжение механизмов терморегуляции, что может снижать общее самочувствие и эффективность труда.

3.2 Экологичность

В настоящее время самым строгим мировым стандартом экологичности для компьютерной техники является стандарт ТСО-99. В отличие от предыдущих версий, он включает дополнительные требования по экологии, эргономике, энергопотреблению и эмиссии устройств. Другие важные маркировки, такие как «Голубой ангел», присуждаемые Германской сертификационной организацией как знак соответствия экологическим стандартам, крайне редки в сфере электроники. Напротив, широко распространен логотип

«Energy Star», которым отмечаются энергосберегающие устройства. Однако проблема с этой маркировкой заключается в том, что производители могут самостоятельно присваивать её своей продукции без обязательной проверки. Поскольку этот знак не дает точной информации о фактическом энергопотреблении устройств, его можно игнорировать.

ЖК-экраны являются источником парниковых газов, которые гораздо опаснее диоксида углерода. Жидкокристаллические мониторы быстро завоевали популярность, вытеснив громоздкие ЭЛТ-модели. Это неудивительно, ведь они имеют тонкий корпус и потребляют значительно меньше электроэнергии. С точки зрения экологической безопасности дисплеи на основе жидких кристаллов считались прорывом, поскольку в их производстве не использовался газ, содержащий свинец. Долгое время никто не обращал внимания на использование для чистки ЖК-панелей трехфтористого азота (NF₃), и только в середине 2008 года ученые обнаружили это вещество в атмосфере. Открытие было поразительным: по сравнению с диоксидом углерода (CO₂), NF₃ является в 17 000 раз более мощным парниковым газом, а его время полураспада в атмосфере может составлять от 550 до 740 лет (у CO₂ - от 30 до 40 лет). Законов, ограничивающих выбросы NF₃, пока не существует.

Определение энергопотребления так же сложно, как и оценка количества перерабатываемых материалов и содержание тяжелых металлов в устройствах. Организация Greenpeace получила удивительные результаты при сравнительном анализе нескольких идентичных моделей ноутбуков из разных стран. Например, в ноутбуке Dell Vostro V13, доступном на китайском рынке, обнаружены следы брома в тачпаде, в то время как в модели из Германии он присутствовал в кнопках. В американской версии этого ноутбука бром обнаружен в блоке питания. Аналогичные отличия замечены и в продукции Apple: кабель ноутбука MacBook Pro 13 из США и Нидерландов содержал в три раза больше брома, чем те же устройства из Филиппин и России. При анализе другого кабеля бром был обнаружен в устройствах из Рос-

сии и Нидерландов, но отсутствовал в американской модели. Таким образом, единственным надежным показателем экологичности остается уровень энергопотребления, и лишь несколько моделей субноутбуков выделяются как лидеры в этом плане.

Производство, занимающееся добычей, причиняет серьезный вред земной поверхности и часто загрязняет окружающий воздух и воду. Извлечение редкоземельных минералов требует применения процессов, которые серьезно наносят ущерб окружающей среде.

Поливинилхлорид, известный как ПВХ, это разновидность пластика, применяемая в различных областях. Он используется для изготовления оболочек кабелей, в которые встроены устройства, а также для изоляции проводов в портативных компьютерах. ПВХ встречается в музыкальных пластинках и применяется для изготовления труб и одежды. Этот материал дешев, прочен и широко распространен. Однако, согласно IT-аналитику из "Гринпис" Кейси Харрелла, ПВХ считается одним из самых вредных видов пластика. Он может вызывать гормональный дисбаланс, проблемы в репродуктивной системе и различные виды рака.

Утилизация ПВХ представляет серьезную проблему. Обычно этот материал либо отправляется на свалку с отходами, либо, что еще хуже, сжигается для извлечения ценных компонентов, таких как медные жилы. При сжигании образуется канцерогенный диоксин. Свалки и химические захоронения загрязняют водные источники. Единственный правильный способ утилизации ПВХ заключается в его отправке в центр опасных отходов.

Утилизация старых компьютеров представляет собой важную практику с точки зрения экологии и управления отходами. Старые компьютеры могут содержать опасные вещества, такие как свинец и ртуть, которые могут загрязнять окружающую среду и представлять угрозу для здоровья. Правильная утилизация помогает избежать таких негативных последствий. Существует несколько способов утилизации старых компьютеров. Один из них -

переработка. Это процесс, в ходе которого компьютер разбирается на составные части, которые затем перерабатываются и используются для производства новых изделий. Другой вариант - пожертвование. Если компьютер все еще работает, его можно передать нуждающимся организациям или школам.

Также многие производители компьютеров предлагают программы по утилизации, позволяющие сдать старое устройство обратно производителю для последующей переработки или утилизации. Некоторые люди выбирают обмен или продажу старых компьютеров для получения дополнительных средств или скидок при покупке нового оборудования. Важно помнить о том, что правильная утилизация старых компьютеров способствует уменьшению отходов и защите окружающей среды. Поэтому, если у вас есть старое устройство, необходимо принять меры для его безопасной и экологически безопасной утилизации.

3.3 Чрезвычайные ситуации

Короткое замыкание:

Наиболее вероятной чрезвычайной ситуацией, при работе с компьютером, является короткое замыкание. Чтобы предотвратить возникновение короткого замыкания необходимо соблюдать следующие меры предосторожности:

- регулярная проверка состояния электропроводки и убедительное обеспечение отсутствия повреждений или изоляционных нарушений;
- использование качественных розеток и удлинителей, адаптированных к предполагаемой нагрузке;
- избежание перегрузки электрических сетей и разумное распределение электроприборов по розеткам;
- своевременное обнаружение и прекращение использования устройств при обнаружении признаков короткого замыкания, таких как дым, запах гари и т. д;

- регулярное техническое обслуживание компьютерной техники, включая очистку от пыли и замену изношенных деталей;
- обучение персонала процедурам безопасности в чрезвычайных ситуациях, включая план эвакуации и использование огнетушителей;
- немедленный вызов спасательных служб и немедленная эвакуация из здания в случае возникновения пожара, с соблюдением всех инструкций по безопасности.

Пожар:

Наиболее вероятная чрезвычайная ситуация, которая может возникнуть при работе с ПЭВМ – пожар, так как в современных ЭВМ очень высокая плотность размещения элементов электронных схем, в непосредственной близости друг от друга располагаются соединительные провода и кабели, при протекании по ним электрического тока выделяется значительное количество теплоты, при этом возможно оплавление изоляции и возникновение возгорания.

Чтобы предотвратить возникновение пожара, необходимо регулярно проверять техническое оборудование и незамедлительно устранять любые неполадки, которые могут привести к возгоранию. После завершения работы следует проводить уборку рабочих мест и помещений, отключать электричество, за исключением дежурного освещения, и обеспечивать надлежащее содержание и постоянную готовность средств пожаротушения, связи и сигнализации.

Для борьбы с пожаром на начальном этапе используются огнетушители. Поскольку в помещениях с компьютерной техникой наиболее вероятными классами пожаров являются «А» и «Е» (то есть могут гореть в основном твёрдые вещества, сопровождающиеся тлением – класс А, или возможны пожары, вызванные возгоранием электроустановок – класс Е), следует применять углекислотные и порошковые огнетушители.

Углекислотный огнетушитель ОУ-5 предназначен для тушения различных материалов, предметов и веществ, а также электроустановок под напряжением до 1 кВ и используется для тушения компьютерной техники и оргтехники. Во время пожара необходимо поднести огнетушитель как можно ближе к огню, направить раструб на очаг пожара, сорвать пломбу (выдернуть чеку), открыть вентиль, нажать на пусковой рычаг и направить струю газа на пламя. Важно помнить, что во время работы раструб нельзя держать рукой, так как он имеет очень низкую температуру.

Порошковый огнетушитель ОП-5 предназначен для тушения твёрдых, жидких, газообразных веществ и электроустановок под напряжением до 1 кВ и используется для тушения компьютерной техники и оргтехники. При пожаре следует поднести огнетушитель к очагу возгорания, выдернуть чеку, нажать на рычаг и направить шланг с распылителем на огонь.

Расстояние от возможного источника возгорания до места установки огнетушителя не должно превышать 20 метров, если компьютеры размещены в общественных зданиях и сооружениях.

В помещениях объёмом до 50 м³ вместо или вместе с переносными огнетушителями можно использовать самосрабатывающие порошковые огнетушители (ОСП и другие).

Чтобы избежать паники и обеспечить быструю и безопасную эвакуацию сотрудников (если возникнет дым в помещении и коридорах), возле дверей, выключателей, рубильников и на путях эвакуации следует разместить фотолюминесцентные знаки. Вычислительные центры должны иметь минимум два отдельных эвакуационных выхода.

Для автоматического обнаружения пожаров можно использовать различные датчики. Главное, чтобы они реагировали на определённые параметры окружающей среды. Ручные датчики предназначены для передачи информации о пожаре через человека и должны быть установлены на высоте

1,5 метра от пола. Автоматические пожарные датчики, кроме световых, размещаются на потолке в помещении.

ЗАКЛЮЧЕНИЕ

В результате выпускной квалификационной работы была разработана и внедрена информационная система для администрации Константиновского района, обеспечивающая эффективное управление данными и контроль над их изменениями. Функционал системы включает добавление, изменение и удаление заявок, что соответствует требованиям заказчика.

Использование современных технологий и методов разработки гарантирует высокое качество продукта. Разработанная система не только удовлетворяет текущие потребности администрации, но и имеет потенциал для дальнейшего развития и улучшения. Это повысит эффективность работы администрации, улучшит процессы управления и сделает их более прозрачными.

Разработанная система не только удовлетворяет текущие потребности администрации, но и имеет потенциал для дальнейшего развития и улучшения. Возможность добавления новых функциональных модулей и адаптации системы под изменяющиеся требования делает её гибким инструментом для решения будущих задач. Это может включать интеграцию с другими государственными системами, внедрение дополнительных аналитических инструментов и расширение возможностей для граждан через онлайн-сервисы.

Результаты проекта значительно повышают эффективность работы администрации, улучшая процессы управления и делая их более прозрачными. Внедрение системы способствует сокращению времени на обработку заявок, улучшению качества обслуживания граждан и повышению удовлетворенности населения.

БИБЛИОГРАФИЧЕСКИЙ СПИСОК

- 1 Анжела, Байрон Drupal. Создание и управление сайтом / Байрон Анжела. - М.: Символ-плюс, 2021. - 242 с.
- 2 Базы данных : учебное пособие для обучающихся по направлению подготовки 09.03.03 «Прикладная информатика» / составители Т. Ж. Базаржапова, О. А. Гармаева, А. Ю. Хаптахаяев. — Улан-Удэ : Бурятская государственная сельскохозяйственная академия им. В.Р. Филиппова, 2022. — 84 с. — Текст : электронный // Цифровой образовательный ресурс IPR SMART : [сайт]. — URL: <https://www.iprbookshop.ru/125200.html>
- 3 Богданов, А.Г. 47 готовых решений для создания Web-сайта и др. - Москва: Гостехиздат, 2021. - 272 с.
- 4 Борисенко А.А. Web-дизайн. Просто как дважды два. - М.: Эксмо, 2008.- 320 с.
- 5 Введение в СУБД MySQL : учебное пособие. — 3-е изд. — Москва : Интернет-Университет Информационных Технологий (ИНТУИТ), Ай Пи Ар Медиа, 2021. — 228 с. — ISBN 978-5-4497-0912-7. — Текст : электронный // Цифровой образовательный ресурс IPR SMART : [сайт]. — URL: <https://www.iprbookshop.ru/102004.html>
- 6 Дакетт, Д. HTML и CSS. Разработка и создание веб-сайтов (+ CD-ROM) / Д. Дакетт. - М.: Эксмо, 2020. - 756 с.
- 7 Джамса Крис. Эффективный самоучитель по креативному Web-дизайну. HTML, XHTML, CSS, JavaScript, PHP, ASP, ActiveX. Текст, графика, звук и анимация. Пер с англ./Крис Джамса, Конрад Кинг, Энди Андерсон - М.: ООО "ДиаСофтЮП", 2005.- 672 с.
- 8 Дэвид, Скляр Изучаем PHP 7. Руководство по созданию интерактивных веб-сайтов / Скляр Дэвид. - М.: Диалектика / Вильямс, 2019. - 337 с.
- 9 Кара-Ушанов, В. Ю. SQL — язык реляционных баз данных : учебное пособие / В. Ю. Кара-Ушанов. — Екатеринбург : УрФУ, 2016. — 156 с. —

ISBN 978-5-7996-1622-9. — Текст : электронный // Лань : электронно-библиотечная система. — URL: <https://e.lanbook.com/book/98296>

10 Китинг Flash MX. Искусство создания web-сайтов / Китинг, Джоди. - М.: ТИД ДС, 2019. - 848 с.

11 Кузнецов PHP. Практика создания Web-сайтов / Кузнецов, М.В. и. - М.: БХВ-Петербург, 2019. - 457 с.

12 Маценко, В.Ф. Имидж: Психология создания / В.Ф. Маценко. - М.: Ника-Центр, 2022. - 112 с.

13 Машнин, Тимур Bootstrap: Быстрое создание современных сайтов / Тимур Машнин. - М.: Издательские решения, 2018. - 253 с.

14 Мерсер Drupal 6. Создание надежных и полнофункциональных веб-сайтов, блогов, форумов, порталов и сайтов-сообществ / Мерсер, Дэвид. - М.: Вильямс, 2022. - 272 с.

15 Местное самоуправление и муниципальное управление / Под ред. А.С. Прудникова, М.С. Трофимова. - М.: Юнити, 2016. - 553 с.

16 Полубояров, В. В. Использование MS SQL Server Analysis Services 2008 для построения хранилищ данных : учебное пособие / В. В. Полубояров. — 3-е изд. — Москва : Интернет-Университет Информационных Технологий (ИНТУИТ), Ай Пи Ар Медиа, 2021. — 662 с. — ISBN 978-5-4497-0883-0. — Текст : электронный // Цифровой образовательный ресурс IPR SMART : [сайт]. — URL: <https://www.iprbookshop.ru/102014.html>

17 Парахина, В.Н. Муниципальное управление / В.Н. Парахина, Е.В. Галеев, Л.Н. Панькова. - М.: КноРус, 2018. - 544 с.

18 Сидорова, Н. П. Базы данных: практикум по проектированию реляционных баз данных : учебное пособие / Н. П. Сидорова. — Королёв : МГОТУ, 2020. — 92 с. — ISBN 978-5-4499-0799-8. — Текст : электронный // Лань : электронно-библиотечная система. — URL: <https://e.lanbook.com/book/149436>

19 Смирнов, М. В. Администрирование баз данных MS SQL Server

2019 : учебно-методические пособия / М. В. Смирнов, Р. С. Толмасов. — Москва : РТУ МИРЭА, 2021. — 98 с. — Текст : электронный // Лань : электронно-библиотечная система. — URL: <https://e.lanbook.com/book/226667>

20 Токмаков, Г. П. Базы данных: Модели и структуры данных, язык SQL, программирование баз данных : учебное пособие / Г. П. Токмаков. — Ульяновск : УлГТУ, 2021. — 362 с. — ISBN 978-5-9795-2184-8. — Текст : электронный // Лань : электронно-библиотечная система. — URL: <https://e.lanbook.com/book/259706>

21 Этан, Браун Изучаем JavaScript. Руководство по созданию современных веб-сайтов / Браун Этан. - М.: Альфа-книга, 2020. - 628 с.

ПРИЛОЖЕНИЕ А

Техническое задание

1. Введение

1.1. Наименование программы – «Электронная приёмная»

1.2. Краткая характеристика области применения – Информационная система «Электронная приёмная» служит инструментом для автоматизации и оптимизации взаимодействия между гражданами и администрацией, а также для повышения эффективности управления запросами и заявками от населения.

2. Основания для разработки

Основанием для разработки служит задание курсовой работе.

3. Назначение разработки

Информационная система будет использоваться следующими пользователями:

- граждане;
- модератор;
- работник администрации.

3.1. Функциональное назначение – Заключается в приеме, обработке и управлении заявками, которые поступают от жителей села. Граждане могут подавать свои заявки в электронном виде через удобный интерфейс, что значительно упрощает процесс взаимодействия с администрацией. После подачи заявки система автоматически назначает ответственных сотрудников для её обработки, отслеживает выполнение и ведет историю всех действий, связанных с заявкой.

3.2. Эксплуатационное назначение – Программа может эксплуатироваться, как на рабочем компьютере оператора, так и на домашнем.

4. Требования к программе или программному изделию

Продолжение ПРИЛОЖЕНИЯ А

4.1. Требования к функциональным характеристикам – Система должна позволять гражданам подавать заявки и обращения через веб-интерфейс, включая возможность прикрепления файлов и документов. Важно, чтобы заявки могли быть автоматически распределены между ответственными сотрудниками, а пользователи могли редактировать и удалять свои заявки при необходимости, при этом вся история изменений должна сохраняться.

Интерфейс системы должен быть интуитивно понятным и удобным для пользователей с различным уровнем компьютерной грамотности. Важно обеспечить круглосуточный доступ к системе, позволяя гражданам подавать и отслеживать свои заявки в любое время.

Все данные хранятся в базе данных.

4.2. Требования к надежности – Вероятность безотказной работы системы должна составлять не менее 99.99%

4.2.1. Требования к обеспечению надежного (устойчивого) функционирования программы – Надежное (устойчивое) функционирование программы должно быть обеспечено выполнением заказчиком совокупности организационно-технических мероприятий, перечень которых приведен ниже:

- а) Использованием лицензионного программного обеспечения;
- б) Регулярным выполнением рекомендаций Министерства труда и социального развития РФ, изложенных в Постановлении от 23 июля 1998 г. «Об утверждении межотраслевых типовых норм времени на работы по сервисному обслуживанию ПЭВМ и оргтехники и сопровождению программных средств»;
- в) Регулярным выполнением требований ГОСТ 51188-98. Защита информации. Испытания программных средств на наличие компьютерных вирусов.

4.3. Условия эксплуатации – Система должна быть развернута на серверном оборудовании, соответствующем требованиям по производительности

Продолжение ПРИЛОЖЕНИЯ А

сти и надежности. Серверы должны быть оснащены современными процессорами, достаточным объемом оперативной памяти и высокоскоростными накопителями данных. Желательно использование серверов с поддержкой отказоустойчивых решений, таких как RAID-массивы и системы резервного копирования.

Для обеспечения круглосуточного доступа пользователей к системе необходимо наличие стабильного и высокоскоростного интернет-соединения. Серверы должны быть подключены к сети с минимальной задержкой и высокой пропускной способностью для обеспечения быстрой обработки запросов.

Для предотвращения потери данных в случае сбоя оборудования или других чрезвычайных ситуаций должна быть настроена система регулярного резервного копирования. Резервные копии должны храниться на отдельном носителе и регулярно проверяться на возможность восстановления.

Особых климатических условий или требований к видам обслуживания не требуется.

Оператор должен обладать практическими навыками работы с графическим пользовательским интерфейсом операционной системы.

4.4. Требования к техническим характеристикам – Компьютер оператора должен быть не хуже следующих характеристик:

Процессор: 1 ГГц или выше

ОЗУ: 2 Гб (для 64-разрядной системы)

Свободное место на жестком диске: 20 Гб (для 64-разрядной системы)

видеокарту, монитор, мышь, клавиатура

5. Требования к программной документации

Предварительный состав программной документации:

– Техническое задание

– руководство оператора

Продолжение ПРИЛОЖЕНИЯ А

6. Техничко-экономические показатели

Автоматизация обработки заявок уменьшает потребность в ручной обработке, сокращая трудозатраты сотрудников администрации. Эффективное распределение заявок и автоматизация рутинных задач освобождают время сотрудников для выполнения других важных задач. Переход на электронную систему уменьшает необходимость в использовании бумаги, снижая расходы на материалы и печать. Упрощение и стандартизация процессов обработки заявок и отчетности позволяет более эффективно управлять ресурсами и снижать операционные расходы.

7. Стадии и этапы разработки

Разработка должна быть проведена в три стадии:

- а) техническое задание
- б) технический (и рабочий) проекты
- в) внедрение

На стадии «Техническое задание» должен быть выполнен этап разработки, согласования и утверждения настоящего технического задания.

На стадии «Технический (и рабочий) проект» должны быть выполнены перечисленные ниже этапы работ:

- а) разработка программы;
- б) разработка программной документации;
- в) испытания программы.

На стадии «Внедрение» должен быть выполнен этап разработки «Подготовка и передача программы».

Содержание работ по этапам:

На этапе разработки технического задания должны быть выполнены перечисленные ниже работы:

- а) постановка задачи;
- б) определение и уточнение требований к техническим средствам;

Продолжение ПРИЛОЖЕНИЯ А

- в) определение требований к программе;
- г) определение стадий, этапов и сроков разработки программы и документации на нее;
- д) согласование и утверждение технического задания.

На этапе разработки программы должна быть выполнена работа по программированию (кодированию) и отладке программы.

На этапе разработки программной документации должна быть выполнена разработка программных документов в соответствии с требованиями ГОСТ 19.101-77.

На этапе испытаний программы должны быть выполнены перечисленные ниже виды работ:

- а) разработка, согласование и утверждение порядка и методики испытаний;
- б) проведение приемо-сдаточных испытаний;
- в) корректировка программы и программной документации по результатам испытаний.

На этапе подготовки и передачи программы должна быть выполнена работа по подготовке и передаче программы и программной документации в эксплуатацию на объектах заказчика.

8. Порядок контроля и приемки

Приемосдаточные испытания программы должны проводиться согласно разработанной исполнителем и согласованной заказчиком.

Ход проведения приемо-сдаточных испытаний заказчик и исполнитель документируют в протоколе испытаний.

На основании протокола испытаний исполнитель совместно с заказчиком подписывают акт приемки-сдачи программы в эксплуатацию.