**СОДЕРЖАНИЕ**

[ВВЕДЕНИЕ 2](#_Toc135069591)

[1 Анализ задачи и формулирование технического задания 3](#_Toc135069592)

[1.1 Вводный инструктаж 3](#_Toc135069593)

[1.2 Организационная структура предприятия 3](#_Toc135069594)

[1.3 Постановка задачи и разработка алгоритма по ее решению 5](#_Toc135069595)

[1.4 Определение состава подсистем и функциональных задач 6](#_Toc135069596)

[1.5 Разработка технического задания в соответствии с ГОСТ ЕСПД 6](#_Toc135069597)

[2 Разработка серверной и клиентской части приложения 8](#_Toc135069598)

[2.1 Визуальное моделирование 8](#_Toc135069599)

[2.2 Разработка модулей программного обеспечения 9](#_Toc135069600)

[2.3 Разработка базы данных 11](#_Toc135069601)

[3 Отладка и тестирование приложения 13](#_Toc135069602)

[4 Оформление документации на программные средства с использованием инстурментальных средств 14](#_Toc135069603)

[4.1 Создание IDEF0 – диаграммы модуля 14](#_Toc135069604)

[4.2 Оформление документации, сопровождающей процесс верификации и тестирования 14](#_Toc135069605)

[4.3 Разработка руководства пользователя 17](#_Toc135069606)

[4.4 Разработка руководства программиста 17](#_Toc135069607)

[ЗАКЛЮЧЕНИЕ 18](#_Toc135069608)

[ПРИЛОЖЕНИЕ А 21](#_Toc135069609)

[ПРИЛОЖЕНИЕ Б 24](#_Toc135069610)

[ПРИЛОЖЕНИЕ В 25](#_Toc135069611)

[ПРИЛОЖЕНИЕ Г 29](#_Toc135069612)

## ВВЕДЕНИЕ

Я, Пятак Ярослав Алексеевич, студент 3 курса группы 20-ПКС-11 проходил производственную (преддипломную) практику в Администрации Староминского сельского поселения, расположенной по адресу – ул. Красная 25. Установленные сроки проведения практики с 20.04.2023 г. по 17.05.2023 г. Назначенный руководитель – Дорофеев Максим Григорьевич.

Целью прохождения производственной (преддипломной) практики является закрепление и углубление теоретических знаний, полученных в процессе обучения в колледже, а также выполнение задач связанных с выпускной квалификационной работой.

Основными задачами производственной (преддипломной) практики по являются:

- проектирование модулей программного обеспечения выпускной квалификационной работы;

- программирование модулей программного обеспечения выпускной квалификационной работы;

- разработка тестовых наборов и тестовых сценариев для модулей программного обеспечения выпускной квалификационной работы;

- оформление практической и теоретической частей выпускной квалификационной работы;

- подготовка письменного отчёта;

# 1 Анализ задачи и формулирование технического задания

## Вводный инструктаж

По прибытие на место прохождения производственной практики был произведен вводный инструктаж, который включал в себя: правила поведения работника на предприятии, устройство и основные задачи предприятия, изучение правил внутреннего распорядка, правила и нормы охраны труда, технику безопасности при работе с вычислительной техникой. На основе полученной информации был проведен анализ с последующим составлением организационной структуры предприятия.

## Организационная структура предприятия

Структуру администрации сельского поселения образуют: глава администрации сельского поселения, заместители главы администрации сельского поселения, а также иные должностные лица администрации сельского поселения. Структура администрации сельского поселения утверждается решением земского собрания по представлению главы администрации сельского поселения. Решением земского собрания об утверждении структуры администрации сельского поселения определяются также должностные обязанности каждого из указанных должностных лиц администрации сельского поселения.

Должностные лица администрации сельского поселения в соответствии с настоящим Уставом, а также в соответствии с решением земского собрания сельского поселения о структуре администрации сельского поселения непосредственно организуют исполнение полномочий, указанных в части 1 настоящей статьи.

Администрация Староминского сельского поселения осуществляет организацию решения вопросов местного значения, исполняет полномочия органов местного самоуправления муниципального Староминского района Краснодарского края, переданные на основе соглашений, заключенных главой сельского поселения, а также решает вопросы исполнения отдельных государственных полномочий, переданных для осуществления органам местного самоуправления сельского поселения, федеральным законом и законом Краснодарского края. Все сведения о полномочиях органа местного самоуправления отображены в Уставе Староминского сельского поселения, принятого от 11 марта 2016 года № 18.4. Данный устав содержит в себе сведения о:

- бюджетных полномочиях администрации

- полномочиях администрации в области коммунально-бытового, торгового обслуживания населения, защиты прав потребителей

- полномочиях администрации в области использования автомобильных дорог, осуществления дорожной деятельности

- полномочия администрации в сфере регулирования земельных, лесных, водных отношений и недропользования

- полномочия администрации в области социально-культурного обслуживания населения, архивного дела и связи

- полномочия администрации в области пожарной безопасности

- муниципальный контроль

Так же в задачи администрации Староминского сельского поселения входит принятие обращений от граждан, по вопросам, входящим в обязанности администрации. Граждане могут обратится с вопросом и получить на него полный ответ, предварительно записавшись на прием. Так же можно задать свой вопрос на сайте, администрации, получив там же ответ. Каждый день администрация получает, отправляет, обрабатывает и хранит огромные объемы информации. Помимо обычных обращений граждан, которые нужно обрабатывать практически непрерывно, администрация так же осуществляет ведение своего сайта. Через сайт граждане могут не только задать вопрос или записаться на прием, но и просмотреть информацию о: утвержденном бюджете на год, торгах, закупках, получить информацию в социальной сфере, просмотреть официальные документы многое другое. Для осуществления всего перечисленного выше, предприятию необходимо иметь собственное сетевое оборудование, с помощью которого и осуществляются вышеперечисленные задачи. Администрация – муниципальное учреждение, а значит входит в состав государственной системы. Это означает полное включение администрации во многие цифровые проекты, деятельность которых включает в себя: обмен, распространение и обработка данных, нужных для функционирования государственных учреждений.

# Постановка задачи и разработка алгоритма по ее решению

Перед тем, как определить мои задачи на производственную (преддипломную) практику, совместно с руководителем практики был произведен разговор, в ходе которого мною были показаны мои наработки по выпускной квалификационной работе. Опираясь на предоставленную мной информацию, а также предыдущий мой опыт прохождения практики на предприятии была поставлена следующая задача – написать модуль , в функционал которого должны входить пересылка и хранение заявок, поступающих от обращающихся в администрацию граждан, между сотрудниками администрации. Основные требования к технической реализации модуля: использование HTML 5 в качестве каркаса сайта и CSS 3 в качестве оформления его внешнего вида. Для взаимодействия модуля с сервером, необходимо использовать PHP 8, серверный код должен содержать функции, предназначенные для работой с базой данных MySQL.

Для решения данной задачи был разработан алгоритм по ее решению, который включает в себя следующие пункты:

* Составление технического задания, с помощью которого будут определены необходимые требования к модулю
* Разработка дизайна модуля и его утверждение руководителем практики
* Доработка базы данных с предыдущей производственной практики под новый модуль
* Программирование модуля
* Тестирование модуля
* Интеграция модуля в систему, разработанную мной на предыдущей производственной практике

## 1.4 Определение состава подсистем и функциональных задач

Сам модуль делится на несколько подсистем, функциональные задачи которых зависят от роли пользователя, зашедшего на сайт. Определение роли пользователя происходит с помощью процесса авторизации. Если была произведена авторизация пользователя, то модуль должен связаться с сервером, найти в базе данных запись о необходимом пользователе, узнать его роль в системе и перенаправить на необходимую страницу.

Для пользователей программного продукта будут предоставлены инструменты по: отправке электронного обращения, с возможностью прикрепить собственный файл, указать тему и суть обращения, просмотр обращения, а также написание на него ответа, просмотр всех поступивших обращений с возможностью определения статуса каждого обращения.

## 1.5 Разработка технического задания в соответствии с ГОСТ ЕСПД

Разработка технического задания состоит из 7 пунктов:

1. Введение

2. Основания для разработки  
 3. Назначение разработки

4. Требование к программе и программному изделию

5. Требования к программной документации

6. Технико-экономические показатели

7. Стадии и этапы разработки

Разработка проводилось в соответствии с ГОСТ, техническое задание можно посмотреть в Приложение А.

# 2 Разработка серверной и клиентской части приложения

## 2.1 Визуальное моделирование

Визуальное моделирование модуля происходило в онлайн редакторе Figma.

Макет будущего модуля должен быть полностью согласован с руководителем практики. Все моделирование происходило в несколько этапов. На первом этапе разрабатывается черновой вариант макета, но так как у меня уже были наработки с размерами и расположениями необходимых элементов, я решил взять их за основу. Утвердив черновой вариант, происходит преобразование макета. На макете расставляются элементы, им придаются необходимые размеры, шрифты и тени. Далее происходит работа с палитрой цветов, а также формами элементов. При моделировании стоит также учитывать то, что диагональ экрана у всех пользователи разная, поэтому стоит заранее задуматься об адаптивности странице. В своем макете я использовал основу с фиксированным размером, который я взял, опираясь на минимальное разрешение современных персональных компьютеров. (см. Рисунок 1).



Рисунок 1 – Финальный вариант макета страницы

# 2.2 Разработка модулей программного обеспечения

Разработка модуля делиться на 2 этапа, разработка Frontend – клиентской части и разработка Backend – серверной части (см. Рисунок 2). Прежде всего был составлен HTML каркас клиентской части, связывание страниц между собой. Тут же, прямо в HTML разметке делается заготовка для серверной части в виде формы, которая будет получать введенные в нее данные, присваивать данным заголовок и отправлять на страницу с PHP скриптом, где эта форма и будет обработана. После чего всем элементам с помощью CSS были добавлены стили и подключен шрифт. Для упрощения верстки, к каждой странице, помимо своего файла CSS, так же подключен файл, отвечающий за обнуление или же нормализацию CSS селекторов. Он отключает стандартные стилевые оформления, которые изначально были заложены в HTML элементы, а также помогает корректно отображать сайт на разных браузерах.

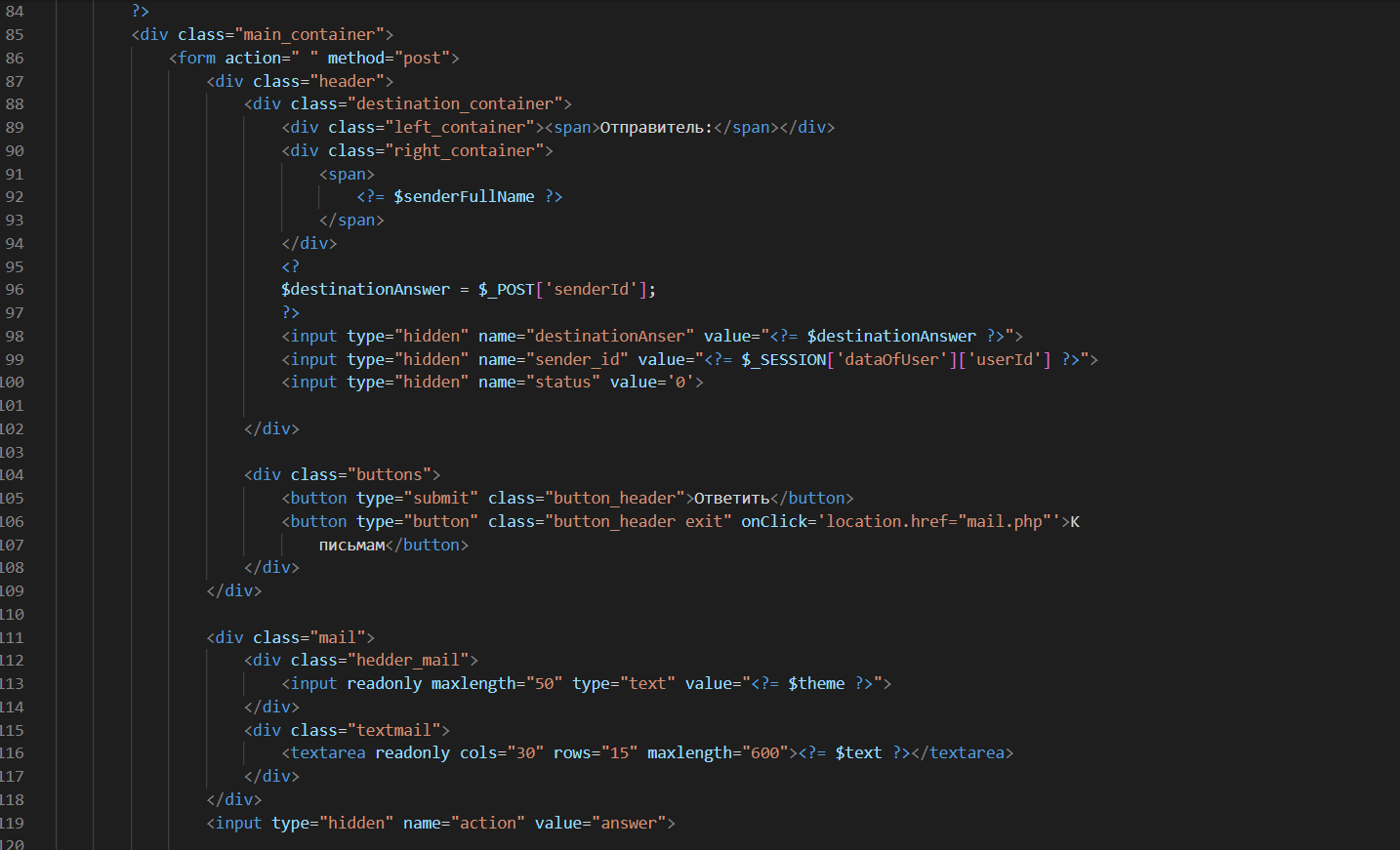


Рисунок 2 – Frontend файл модуля

Серверная часть модуля состоит из PHP скриптов, базы данных, а также сервера, который будет хранить и обрабатывать все данные. Так как мне было порученно разработать модуль для сайта, то и сервер и база данных уже были настроены и готовы к подключению модуля. Прежде всего, необходим PHP скрипт, который будет хранить адрес подключения к базе данных (см. Рисунок 3). Так как почти каждый файл нуждается к подключению к базе данных, то скрипт с подключением был вынесен в отдельный файл, который записывал в переменную адрес базы данных, а также предусматривал случай, невозможного подключения, в следствии чего выводил ошибку пользователю. Написав скрипт подключения один раз, далее необходимо просто связать все файлы с данным скриптом, и код будет выполняться в отдельном файле.

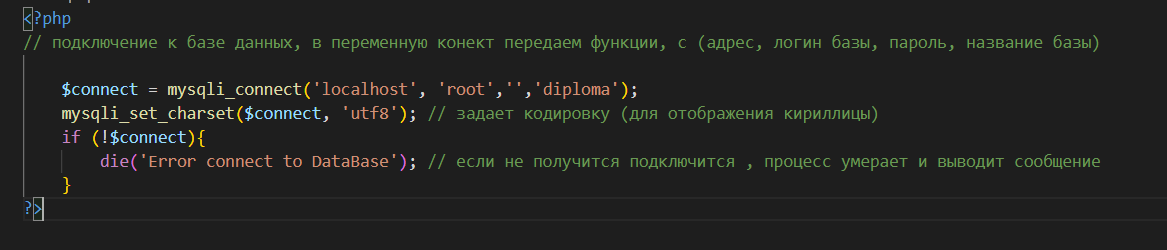


Рисунок 3 – PHP скрипт подключения к базе данных

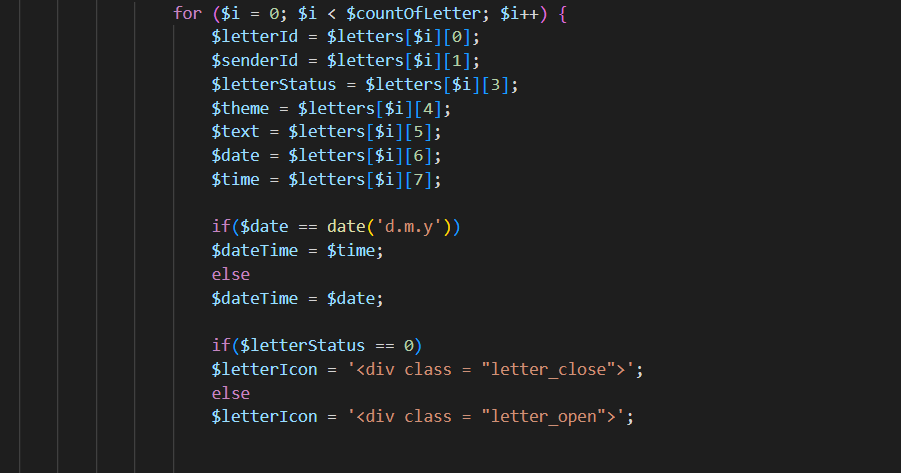
Далее необходимо заняться отправкой данных, введеных пользователем. Для этого были использованы формы. Каждая форма содержит в себе видимые поля ввода, куда пользователи вводят свои данные, а также скрытое поле ввода, которое передает тип дейсвия этой формы. После того, как форма будет отправлена, данные передаются скрытым методом post на php страницу, которая содержит скрипт, обрабатываемый полученные данные. Скрипт получает данные и взависимости от условий заполняет таблицы в базе данных (см. Рисунок 4). 

Рисунок 4 – PHP скрипт, который обрабатывает полученные данные

# 2.3 Разработка базы данных

Несмотря на то, что база данных для работы с обращениями была разработана мной на предыдущей производственной практике, необходимо доработать ее и подготовить необходимые таблицы. Для безопасной работы с базой данных, была сделана ее копия с сервера, которая редактировалась на локальном сервере OpenServer (см Рисунок 5).



Рисунок 5 – дополнительная таблица, необходимая для работы модуля

Перед тем, как отправлять готовую базу данных необходимо протестировать ее на локальном сервере, проверить правильно ли расставлены типы данных, а также совпадает ли кодировка с базой данных не сервере. После проделанной работы формируется SQL файл создания только нужных таблиц и через импорт файла загружается на сервер (см. Рисунок 6).

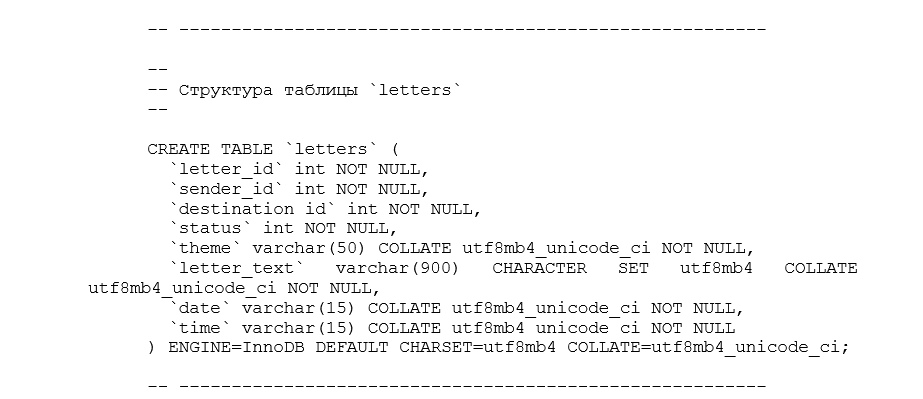


Рисунок 6 – SQL файл, создающий таблицу

# 3 Отладка и тестирование приложения

Тестирование данного модуля происходило в несколько этапов. На первом этапе с помощью инструментов разработчиков, встроенных в браузер google chrome было протестировано поведение модуля при различном разрешении монитора пользователя (см. Рисунок 7).

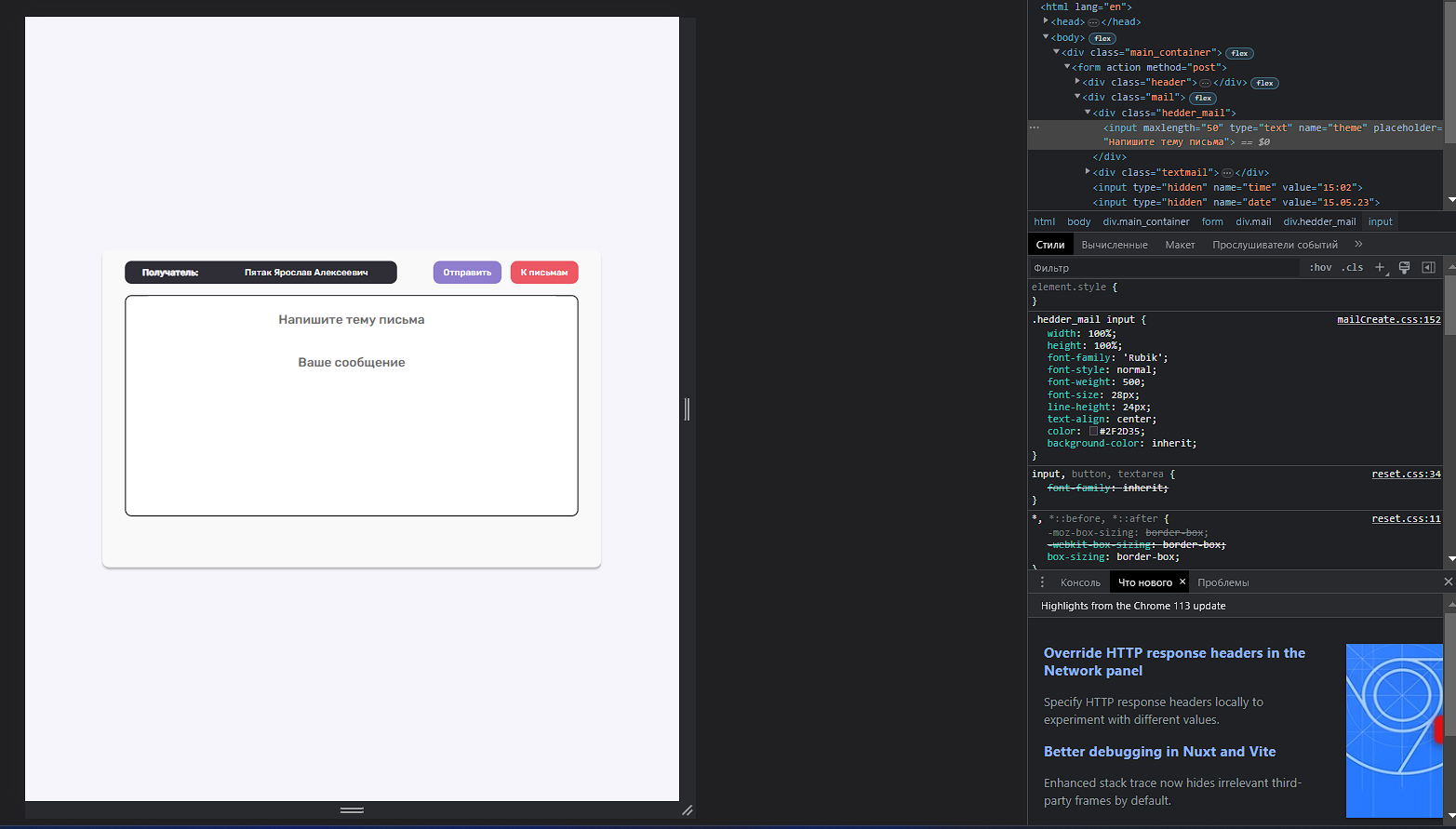


Рисунок 7 – Проверка адаптивности страницы

Далее происходит тестирование работы всех скриптов и полей ввода формы. В первую очередь, простым вводом значений, проверяются поля ввода. Если у тегов HTML стоят изначально правильные атрибуты, то форма не даст ввести пользователю лишнее или неподходящие символы. Для тестирования PHP скриптов необходим локальный сервер, способный запускать эти скрипты и взаимодействовать с базой данных. Далее происходит тестирование PHP скриптов. Делается это обычным заполнением формы и ее отправкой на сервер. Если в базе данных появится заполненная ранее форма, то тестирование формы прошло успешно. В ином же случае сам скрипт выведет ошибку с укзанием файла и строчки, в которых она содержиться.

# 4 Оформление документации на программные средства с использованием инстурментальных средств

## 4.1 Создание IDEF0 – диаграммы модуля

IDEF0 — методология функционального моделирования (англ. function modeling) и графическая нотация, предназначенная для формализации и описания бизнес-процессов. Отличительной особенностью IDEF0 является ее акцент на соподчиненность объектов. В IDEF0 рассматриваются логические отношения между работами, а не их временная последовательность (поток работ). Для создания диаграммы был использован онлайн сервис creately (см. Рисунок 8).

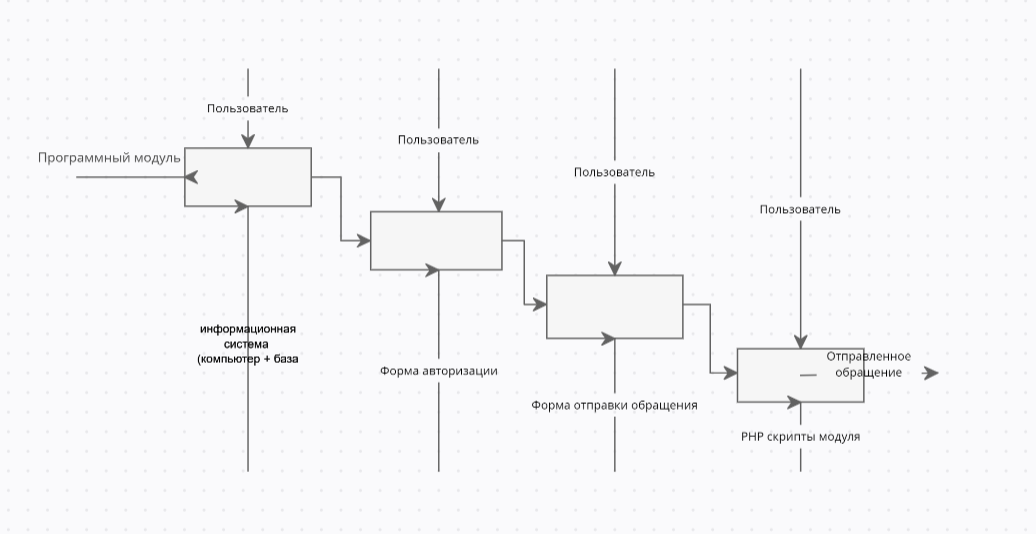


Рисунок 8 – IDEF0 – диаграмма модуля

## 4.2 Оформление документации, сопровождающей процесс верификации и тестирования

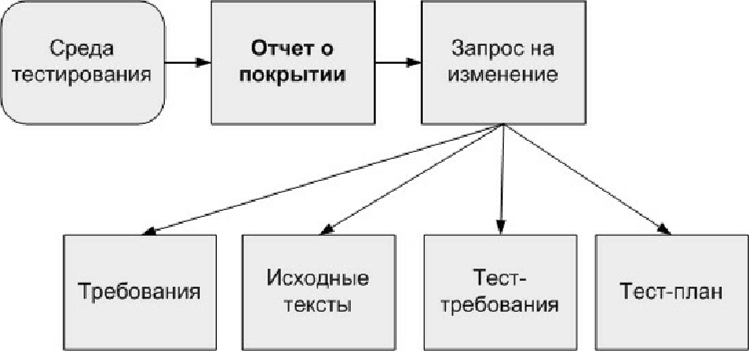
Степень покрытия программного кода тестами – важный количественный показатель, позволяющий оценить качество системы тестов, а в некоторых случаях - и качество тестируемой программной системы. Данные о степени покрытия помещаются в отчеты о покрытии, которые генерируются при выполнении тестов инструментальными средствами, поддерживающими процесс тестирования, т.е. по сути, генерируются средой тестирования (см. Рисунок 9). Формат отчетов о покрытии обычно единый внутри проекта или нескольких проектов и часто зависит от особенностей инструментальных средств тестирования. В отчете о покрытии в стандартизированной форме указываются участки программного кода тестируемой системы (или ее части), которые не были выполнены во время выполнения тестовых примеров, т.е. не были покрыты тестами. Причины непокрытая анализируются тестировщиками, по результатам анализа составляются отчеты о проблемах и запросы на изменение - документы, где описываются объекты разработки, которые необходимо изменить, и причины этих изменений. 

Рисунок 9 – Генерация отчета о покрытии

Недостаточное покрытие может свидетельствовать о неполноте системы тестов или тест-требований, в этом случае в запросе об изменении указывается на необходимость расширения системы тестов или тест-требований. Другой причиной недостаточного покрытия могут быть участки защитного кода, которые никогда не выполнятся даже в случае нештатной работы системы. В этом случае в запросе на изменения указывается на необходимость модификации исходных текстов либо отмечается, что для этого участка программной системы не требуется покрытие. В качестве третьей причины недостаточного покрытия может выступать рассогласование требований и программного кода системы, в результате которого в коде могут остаться неиспользуемые более участки либо, наоборот, появиться участки, рассчитанные на будущее (и реализующие функциональность, не описанную в требованиях). В этом случае в запросе на изменение указывается на необходимость модификации требований и/или кода системы для приведения их в согласованное состояние. Отчет о тестировании модуля можно посмотреть в приложении Б.

## 4.3 Разработка руководства пользователя

Ежедневно создаются новые продукты, программы, сервисы и часто пользователям приходится нелегко при освоении какой-нибудь новой программы или сайта, поэтому каждому новому продукту необходимо собственное руководство пользования. При проведении тестов, стало понятно, что тот интерфейс, который разработчик считает интуитивно понятными, не всегда является таковым для конечного пользователя. Для устранения данной проблемы было составлено руководство пользователя, которое включает в себя инструкции для осуществления всех возможных как для обращающегося гражданина, так и для сотрудника администрации. Для каждого пользователя на сайте размещено свое руководство пользователя, чтобы не допустить донесения до пользователя ненужной ему информации. Руководство пользователя можно посмотреть в Приложение В.

## 4.4 Разработка руководства программиста

Программный продукт, написанный за период производственной практики – модуль, а значит, что этот модуль будет встроен в основную систему сайта администрации. Для дальнейшего поддержание эксплуатации этого модуля необходимо руководство программиста. Благодаря нему, разработчики, которые не участвовали в разработке модуля, смогу ознакомится с версткой модуля, правилами оформления стилей, узнаю о предназначении файлов, а также могут получить подробное описание тех скриптов и тех функций, которые присутствуют в программном продукте. Руководство программиста можно посмотреть в приложении Г.

# ЗАКЛЮЧЕНИЕ

В процессе прохождения производственной (преддипломной) практики были закреплены теоретические знания, приобретены практические навыки и профессиональные компетенции по специальности «программирование компьютерных систем».

Практика началась с вводного инструктажа, изучения требований к организации определённого рабочего места, ознакомления с правилами поведения на предприятии и безопасностью работы.

Далее осуществлялась работа, связанная с разработкой и администрированием баз данных, написанием web-приложения, а также осуществление работы по написанию сопровождающей документации.

Последующие дни практики были посвящены оформлению теоретической части выпускной квалификационной работы. Бали закреплены знания работы с PHP, OpenServer, а так же с Visual Studio Code и работа с технической документацией.

Сделаны следующие выводы: несмотря на всю жесткость требований к выполнению работы в муниципальных и государственных органах, работа может быть интересной и самое главное приносящей пользу огромному числу людей. После прохождения производственной практики, появилось желание изучать и дальше данную область, чтобы расширить профессиональный кругозор и подчерпнуть для себя много новых и полезных знаний.

**СПИСОК ИСПОЛЬЗОВАННЫХ ИСТОЧНИКОВ**

1. Советов, Б. Я. Базы данных : учебник для вузов / Б. Я. Советов, В. В. Цехановский, В. Д. Чертовской. — 3-е изд., перераб. и доп. — Москва : Издательство Юрайт, 2023. — 420 с. — (Высшее образование). — ISBN 978- 5-534-07217-4. — Текст : электронный // Образовательная платформа Юрайт [сайт]. — URL: https://urait.ru/bcode/510752 (дата обращения: 01.03.2023).

2. Полуэктова, Н. Р. Разработка веб-приложений : учебное пособие для вузов / Н. Р. Полуэктова. — Москва : Издательство Юрайт, 2023. — 204 с. — (Высшее образование). — ISBN 978-5-534-13715-6. — Текст : электронный // Образовательная платформа Юрайт [сайт]. — URL: https://urait.ru/bcode/519714 (дата обращения: 12.03.2023).

3. Тузовский, А. Ф. Проектирование и разработка web-приложений : учебное пособие для среднего профессионального образования / А. Ф. Тузовский. — Москва : Издательство Юрайт, 2023. — 218 с. — (Профессиональное образование). — ISBN 978-5-534-10017-4. — Текст : электронный // Образовательная платформа Юрайт [сайт]. — URL: https://urait.ru/bcode/517539 (дата обращения: 15.03.2023).

4. Маркин, А. В. Программирование на SQL : учебное пособие для среднего профессионального образования / А. В. Маркин. — Москва : Издательство Юрайт, 2021. — 435 с. — (Профессиональное образование). — ISBN 978-5-534-11093-7. — Текст : электронный // Образовательная платформа Юрайт [сайт]. — URL: https://urait.ru/bcode/476040 (дата обращения: 31.06.2023).

5. Советов, Б. Я. Базы данных : учебник для вузов / Б. Я. Советов, В. В. Цехановский, В. Д. Чертовской. — 3-е изд., перераб. и доп. — Москва : Издательство Юрайт, 2022. — 420 с. — (Высшее образование). — ISBN 978- 5-534-07217-4. — Текст : электронный // Образовательная платформа Юрайт [сайт]. — URL: https://urait.ru/bcode/488866 (дата обращения: 10.12.2022). 40

6. Маркин, А. В. Программирование на SQL в 2 ч. Часть 1 : учебник и практикум для вузов / А. В. Маркин. — 2-е изд., перераб. и доп. — Москва : Издательство Юрайт, 2022. — 403 с. — (Высшее образование). — ISBN 978- 5-534-12256-5. — Текст : электронный // Образовательная платформа Юрайт [сайт]. — URL: https://urait.ru/bcode/491238 (дата обращения: 10.12.2022).

7. Маркин, А. В. Программирование на SQL в 2 ч. Часть 2 : учебник и практикум для вузов / А. В. Маркин. — 2-е изд., испр. и доп. — Москва : Издательство Юрайт, 2022. — 340 с. — (Высшее образование). — ISBN 978- 5-534-12258-9. — Текст : электронный // Образовательная платформа Юрайт [сайт]. — URL: https://urait.ru/bcode/490104 (дата обращения: 10.12.2022).

# ПРИЛОЖЕНИЕ А

**Техническое задание**

1 Введение

1.1 Название программы

Система написания, обмена и хранения обращений граждан онлайн

1.2 Краткая характеристика области применения

Система написания, обмена и хранения обращений граждан онлайн

будет внедрена на сайт администрации Староминского сельского поселения, как отдельный программный модуль.

2. Основания для разработки

Основанием для разработки является необходимость администрации Староминского сельского поселения в приеме и обработке обращений граждан онлайн.

3. Назначение разработки

3.1. Функциональное назначение

Для пользователей программного продукта будут предоставлены инструменты по: отправке обращения, прикрепления к нему файлов, просмотру ответов на обращения. Для всех пользователей системы будет предоставлен визуальный интерфейс в виде страницы сайта.

3.2. Эксплуатационное назначение

Информационная система должна использоваться на сайте администрации Староминского сельского поселения ее сотрудниками, а также гражданами поселения.

4. Требование к программе и программному изделию

4.1. Требование к функциональным характеристикам

При запуске программного модуля веб-сайта пользователи попадают на главную страницу модуля. Функционал данной страницы будет зависеть от роли пользователя, которая определяется наличием запроса из базы данных. Пользователи могу отправить сообщение необходимому человеку из списка, прикрепить к обращению файл или ответить на уже поступившее обращение.

4.2. Требование к надежности

Вероятность безотказной работы системы должна стремиться к наивысшим показателям при условии исправности сети и оборудования. Рекомендуется хранить одну копию программного обеспечения на внешнем носителе. Также рекомендуется регулярно копировать файл базы данных на внешний носитель, если такую услугу не предоставляет хостинговая компания. Так же он должен выдерживать достаточно хорошие нагрузки.

4.3. Требования к составу и параметрам технических средств

Состав технических средств персонального компьютера пользователя, включает в себя:

процессор x86 с тактовой частотой не менее 1 ГГц;

оперативную память объемом, не менее 1 Гб;

видеокарту, монитор, мышь, клавиатура;

Компьютер администратора, включающий в себя:

процессор x86 с тактовой частотой, не менее 1 ГГц;

оперативную память объемом, не менее 1 Гб;

видеокарту, монитор, мышь, клавиатура;

4.4. Специальные требования

Программа должна обеспечивать взаимодействие с пользователем посредством графического интерфейса, разработанного согласно требованиям компании.

Стадии и этапы разработки

1. Изучение предметной области;

2. Разработка технического задания;

3.Создание модулей и диаграмм

4.Разработка ER-модели

5. Макетирование;

6. Разработка дизайна сайта;

7. Разработка архитектуры СУБД;

8. Создание реляционной СУБД;

9. Разработка клиентской части сайта;

10. Разработка серверной части сайта;

# ПРИЛОЖЕНИЕ Б

**Отчет о тестировании модуля**

**Наименование**

Не работает добавление вводимой информации в форму в базу данных

**Последовательность действий**

1. Зайти в меню выбора действий под ролью обращающегося гражданина
2. Выбрать вкладку «Написать письмо»
3. Заполнить форму обращения
4. Нажать кнопку отправить

**Ожидаемый результат:** Добавление введённых данных в форму в базу данных

**Фактический результат:** Добавление введённых данных в форму не происходит

**Категория дефекта**

Функциональность

**Критичность**

Critical

**Приоритет**

High

**Скриншот**

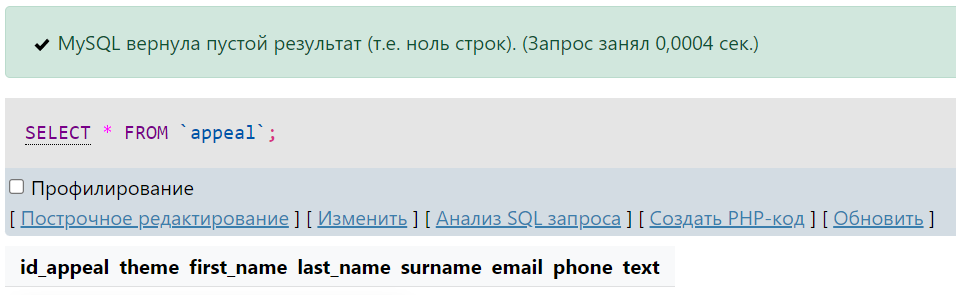
****

Рисунок Б.1 – Скриншот ошибки, при добавлении информации в БД

# ПРИЛОЖЕНИЕ В

**Руководство пользователя**

ИНФОРМАЦИОННАЯ СИСТЕМА

ОПИСАНИЯ ФУНКЦИОНАЛА

«Система написания, обмена и хранения обращений граждан онлайн»

РАБОЧАЯ ДОКУМЕНТАЦИЯ

Руководство пользователя

Листов - 3

Староминская, 2023

Аннотация

Настоящий документ представляет собой руководство пользователя (далее Руководство) информационная система описания функционала «Электронный портал для обращения граждан в администрацию Староминского сельского поселения»

Пользовательский интерфейс электронного портала обеспечивает информационную поддержку деятельности администратора.

Руководство определяет порядок регистрации поставщиков услуг, информационные системы которых осуществляют взаимодействие с администратором, а также порядок формирования отчетности для поставщиков услуг.

Перед работой пользователя с «Система написания, обмена и хранения обращений граждан онлайн» рекомендуется внимательно ознакомиться с настоящим руководством.

Документ подготовлен в соответствии с РД 50-34.698-90 - в части структуры и содержания документов, и в соответствии с ГОСТ 34.201-89 - в части наименования и обозначения документов.

1. Введение для новичков администраторов

1.1. Область применения

Пользовательский интерфейс обеспечивает информационную поддержку деятельности администратора «Система написания, обмена и хранения обращений граждан онлайн» при выполнении следующих операций:

Авторизуется в личный кабинет;

1.2. Краткое описание возможностей

«Система написания, обмена и хранения обращений граждан онлайн»

обеспечивает выполнение следующих основных функций:

Отправка обращений с прикрепление файлов выбранному человеку

Возможность дать ответ на обращение

Просмотр полученных ответов и прикрепленных к ним файлов;

1.3. Уровень подготовки пользователя

Для эксплуатации «Система написания, обмена и хранения обращений граждан онлайн» определены следующие роли:

Сотрудник администрации (администратор)

Администратор должен:

Иметь общие сведения о системе и ее назначении;

Владеть информацией об «Система написания, обмена и хранения обращений граждан онлайн»

в объеме эксплуатационной документации;

Владеть информацией о работе в интерфейсе «Система написания, обмена и хранения обращений граждан онлайн»;

Обеспечивать поддержку взаимодействия с внешними участниками «Система написания, обмена и хранения обращений граждан онлайн»

1.4. Назначение

«Система написания, обмена и хранения обращений граждан онлайн» позволяет:

обеспечить взаимодействие с информационными системами обращающихся граждан и сотрудников администрации;

повысить прозрачность деятельности в сфере оказания услуг за счет формирования полной, достоверной и актуальной информации о ходе и результатах предоставления государственных и иных услуг;

повысить эффективность контроля и анализа процесса предоставления услуг и фактов оплаты оказанных услуг.

1.5. Условия применения

«Система написания, обмена и хранения обращений граждан онлайн»

может эксплуатироваться и выполнять заданные функции при соблюдении требований предъявляемых к техническому, системному и прикладному программному обеспечению.

1.6. Функционал администратора

В данном разделе приводится описание всех операций, существующих в «Система написания, обмена и хранения обращений граждан онлайн»

Таблица В. 1 Функционал администратора

|  |  |  |
| --- | --- | --- |
| Роль | Доступные пункты меню | Доступные операции |
| Администратор | Работа с обращениями граждан | Просмотр  Возможность ответить |
| Выход | Выход из «Система написания, обмена и хранения обращений граждан онлайн» |

# 

# ПРИЛОЖЕНИЕ Г

**Руководство Программиста**

ИНФОРМАЦИОННАЯ СИСТЕМА

РУКОВОДСТВО ПОЛЬЗОВАНИЯ

«Система написания, обмена и хранения обращений граждан онлайн»

РАБОЧАЯ ДОКУМЕНТАЦИЯ

Руководство программиста

Листов - 5

Староминская, 2023

**Условия применения программы**

**Требования к техническим средствам**

Требования к техническим средствам ИС ««Система написания, обмена и хранения обращений граждан онлайн».

Для работы Портала необходимы два высокопроизводительных сервера с конфигурацией не ниже:

Proliant DL380 G5 5160 (Rack2U XeonDC 3.0Ghz(4Mb/) 2х1Gb/P400(256Mb/RAID5/1/0)/noHDD(8)SFF/noCD.noFDD/iLO2std/2xGigEth

Proliant DL380 G5 X5160 (3.00GHz-1x4Mb) Processor Option Kit;

Slimline CD-RW/DWD-ROM Combo Option Kit;

HP 72 Gb SFF SAS 10k rpm Hot Plug Hard Drive (2,5”);

Redundant Power Supply 350/370/380 G5 Worldwide Kit.

**Требования к общему программному обеспечению (ОПО)**

Требования к общему программному обеспечению (ОПО), необходимому для ИС «Единый портал внешнеэкономической информации Минэкономразвития России», представлены в таблице 2.

Таблица 1.2. Требования к ОПО

|  |  |
| --- | --- |
| Наименование | Кол-во |
| 1. ОС Debian Server | 1 |
| 2. PostgreSQL 8.4 | 1 |
| 3. Программные зависимости | 1 |

**Характеристика программы**

Система рассчитана на круглосуточный режим работы в течение 365 дней в году.

Средствами контроля и мониторинга являются автоматизированные встроенные в платформу программные компоненты, которые самостоятельно выполняют следующие действия в случае ошибки

1. Система автоматически перезапускает себя в случае некритической ошибки
2. Посылает администратору сообщение об ошибке в письме на электронную почту
3. Записывают ошибку в журнал, который находится в файловой системе в файле /var/log/messages
4. В случае невозможности совершить вышеуказанные действия, система выведет администратору ошибки прямо в браузер с конкретными данными о том, где, почему и что произошло.

Для восстановления системы необходимо обратиться к документу «Руководство администратора», в котором находится раздел «Восстановление системы после сбоя»

**Внесение  изменений в инсталляцию**

Внесение изменений  в работающую версию будет принято  после перезапуска сервера. Перезапуск сервера может быть осуществлен  как с остановкой HTTP-сервера и прерыванием соединений, так и без них.

Полная перезагрузка HTTP-сервера nginx в среде Linux выполняется  с помощью команды /etc/init.d/nginx restart. Для мягкой перегрузки программной  среды, необходимо выполнить команду touch ~/tmp/restart.txt.

Полная перезагрузка требуется в крайне редких случаях  и в условиях эксплуатации к использованию  рекомендована мягкая перегрузка.

**Процедура автоматизированного  внедрения изменений**

Для автоматизации  внесения изменений при активной разработки, в поставке сайта содержатся настройки утилиты Capistrano. Для автоматизации обновления при постоянных изменениях ее можно использовать без перенастройки со следующим окружением:

• nginx

• mod\_passenger

• ssh-ключ для выкладывающего пользователя

• система контроля версий Git

Для выкладывания необходимо использовать команду cap deploy. Основные настройки выкладывания расположены  в файле ~/config/deploy/staging.rb.

Их описание:

set :repository, "svn+ssh://"

Расположение  репозитория Git.

set :deploy\_to, "/var/www/#{application}"

Путь расположения на сервере.

set :user, "deployer"

Имя пользователя, от лица которого Capistrano будет действовать  на сервере. Этот пользователь должен иметь возможность получить исходные коды из системы контроля версий и быть доступным тому, кто запускает команду cap deploy.

server "[rus-export.dv-t.net](http://forseti.matrixtelecom.net/)", :app, :web

server "[rus-export.dv-t.net](http://forseti.matrixtelecom.net/)", :db, :primary => true

Имена серверов для программного окружения и  СУБД.

**Сообщения**

В ходе выполнения программы, программисту выдаются сообщения. Здесь приводится описание их содержания и действий, которые необходимо предпринять по этим сообщениям.

**Возможные ошибки, которые могут появиться при заходе на веб-страницу единого портала:**  
*dispatch.fcgi  
505 Rails application failed to start*

Решение: мягкая перезагрузка http-сервера

**Возможные ошибки в журнале ошибок (файл /var/log/messages в файловой системе сервера):**

*[warn] mod\_fcgid: Read data error, fastcgi server has close connection*

*[error] [client xxx.xxx.xxx.xxx] Premature end of script headers: dispatch.fcgi*

Решение: мягкая перезагрузка http-сервера

Возможные ошибки в любом из мест:

*PGSQL: Connection pool ended*

*IO: Too much open files*

Решение: необходимо ввести в терминале сервера команду *fs.filemax=16452*

**ПЕРЕЧЕНЬ СОКРАЩЕНИЙ**

|  |  |
| --- | --- |
| ППО | Прикладное программное обеспечение |
| ИС | Информационная система «Единый портал внешнеэкономической информации Минэкономразвития России» |
| СУБД | Система Управления Базами Данных |