МИНИСТЕРСТВО НАУКИ И ВЫСШЕГО ОБРАЗОВАНИЯ РОССИЙСКОЙ ФЕДЕРАЦИИ

Федеральное государственное бюджетное образовательное учреждение высшего образования

«Забайкальский государственный университет»

(ФГБОУ ВО «ЗабГУ»)

Энергетический факультет

Кафедра информатики, вычислительной техники и прикладной математики

КУРСОВОЙ ПРОЕКТ

по дисциплине: Технология разработки программного обеспечения

на тему «WEB-приложение для организации краеведческого туризма»

Выполнил ст. гр. ИВТ-20,

Борисова Е.О.

Проверил доцент, к.т.н., доцент

Валова О.В.

Чита

2024

МИНИСТЕРСТВО НАУКИ И ВЫСШЕГО ОБРАЗОВАНИЯ РОССИЙСКОЙ ФЕДЕРАЦИИ

Федеральное государственное бюджетное образовательное учреждение высшего образования

«Забайкальский государственный университет»

(ФГБОУ ВО «ЗабГУ»)

Энергетический факультет

Кафедра информатики, вычислительной техники и прикладной математики

ПОЯСНИТЕЛЬНАЯ ЗАПИСКА

к курсовому проекту

по 09.03.01 Информатика и вычислительная техника

на тему «WEB-приложение для организации краеведческого туризма»

Выполнил студент группы ИВТ-20 Борисова Екатерина Олеговна

Руководитель работы: доцент, к.т.н., доцент Валова Ольга Валерьевна.

Чита

2024

МИНИСТЕРСТВО НАУКИ И ВЫСШЕГО ОБРАЗОВАНИЯ РОССИЙСКОЙ ФЕДЕРАЦИИ

Федеральное государственное бюджетное образовательное учреждение высшего образования

«Забайкальский государственный университет»

(ФГБОУ ВО «ЗабГУ»)

Энергетический факультет

Кафедра информатики, вычислительной техники и прикладной математики

ЗАДАНИЕ

на курсовой проект

по дисциплине: Технология разработки программного обеспечения

Студенту Борисовой Екатерине Олеговне

направления подготовки 09.03.01 Информатика и вычислительная техника

1. Тема курсового проекта: WEB-приложение для организации краеведческого туризма
2. Срок подачи студентом законченной работы: 03.04.2023 г.
3. Исходные данные к проекту: согласно техническому заданию
4. Перечень подлежащих разработке в курсовом проекте вопросов:
5. Постановка и анализ задачи;
6. Анализ данных;
7. Программная реализация;
8. Техническое задание;
9. Руководство пользователя.
10. Перечень графического материала: -

Дата выдачи задания 06.02.2023 г.

Руководитель курсового проекта \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_О.В.Валова

Задание принял к исполнению

«06» февраля 2023 г.

Подпись студента \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ / Е.О.Борисова/

РЕФЕРАТ

Пояснительная записка – 32 с, 20 рис., 4 таб., 7 источников

КРАЕВЕДЧЕСКИЙ ТУРИЗМ, WEB-СЕРВИС SQLite, Laravel, HTML, CSS

Цель проекта состоит в том, чтобы предоставить удобную и информативную платформу для познания природных достопримечательностей края, поощрить взаимодействие и обмен опытом между посетителями, а также помочь пользователям планировать и организовывать свои поездки.

Программа предназначена для облегчения поиска интересных для посещения мест и компании для путешествия.

СОДЕРЖАНИЕ

[ВВЕДЕНИЕ 7](#_Toc131684732)

[1 Постановка и анализ задачи 8](#_Toc131684733)

[1.1 Описание предметной области 8](#_Toc131684734)

[1.2 Постановка задачи 9](#_Toc131684735)

[1.3 Обоснование выбора средств реализации 9](#_Toc131684736)

[2 Анализ данных 12](#_Toc131684737)

[3 Программная реализация 13](#_Toc131684738)

[4 Техническое задание 18](#_Toc131684739)

[5 Руководство пользователя 21](#_Toc131684740)

[ЗАКЛЮЧЕНИЕ 30](#_Toc131684741)

[СПИСОК ИСПОЛЬЗУЕМЫХ ИСТОЧНИКОВ 31](#_Toc131684742)

ВВЕДЕНИЕ

Краеведческий туризм – это уникальный вид туризма, который позволяет путешественникам погрузиться в историю и культуру мест, которые посещают. В отличие от обычного туризма, краеведческий туризм подразумевает изучение и понимание местных традиций, обычаев и достопримечательностей, посещение природных достопримечательностей.

В нашем крае нет единого источника, в котором были бы указаны возможные для посещения места, о многих из них жители нашего края даже и не знают. Таким образом онлайн-гид, на котором можно будет узнать об этих местах, посмотреть маршрут, найти попутчиков, быть может выбрать турагентство, которое предложит многодневный поход или конную прогулку, будет очень актуален и востребован.

WEB-приложение делится на две составляющие: фронтенд и бэкенд. Для фронтенда используются: язык верстки HTML, язык декодирования и описания внешнего вида веб-страницы CSS, язык программирования JavaScript. Для бэкенда было принято решение использовать бесплатный фреймворк для быстрой разработки веб-приложений Laravel основанный на языке PHP.

На данный момент реализован сайт, на котором можно просмотреть информацию о туристических местах с возможностью комментирования и сохранения в личный кабинет.

**1** **Постановка и анализ задачи**

* 1. **Описание предметной области**

Информационный портал представляет собой онлайн-ресурс, предназначенный для ознакомления пользователей с различными природными достопримечательностями конкретного края (региона). Пользователи могут зарегистрироваться на портале и получить доступ к разнообразной информации о местных природных объектах, таких как национальные парки, заповедники, горы, водопады, озёра и т.д.

Возможности портала включают:

* Каталог достопримечательностей: Пользователи могут просматривать информацию о различных природных объектах края, включая фотографии, описания, географические координаты и отзывы других пользователей.
* Рейтинг и отзывы: Зарегистрированные пользователи могут оставлять свои оценки и отзывы о посещенных достопримечательностях, что поможет другим пользователям с выбором и планированием поездок.
* Группы посещения: Пользователи имеют возможность создавать группы для посещения конкретной достопримечательности или присоединяться к уже созданным группам. Группы могут быть открытыми или закрытыми, в зависимости от предпочтений пользователей.
* Планирование поездок: Пользователи могут создавать планы посещения достопримечательностей, добавлять их в свой профиль, устанавливать даты и оповещения, чтобы быть в курсе предстоящих поездок и активностей в группах.
* Рекомендации и персонализированный контент: Портал может предлагать рекомендации пользователю на основе его предпочтений и предыдущих посещений. Это может включать рекомендации по достопримечательностям, группам посещения или связанным мероприятиям.
* Панель редактора: Редакторы портала имеют возможность управлять контентом, модерировать отзывы и управлять пользователями и группами.
* Панель модератора: Модераторы портала имеют возможность модерировать отзывы и управлять пользователями и группами.

Примером аналогов могут послужить веб-сайты MaxPlant [1] и Узнай Москву [2].

На веб-сайте MaxPlant представлены различные пешие маршруты Санкт- Петербургу и Ленинградской области. На нем пользователь может посмотреть выбрать маршрут, который он хочет посетить. Далее пользователю будет показан маршрут, представленный на Яндекс.Карте и фотографии достопримечательностей с этого маршрута.

Веб-сайт Узнай Москву предоставляет тот же функционал, что и MaxPlant. Дополнительно пользователь может посмотреть каждую достопримечательность отдельно, прочитать историческую справку и оставить комментарий.

**1.2 Диаграмма Вариантов использования**

На Рисунке 1.1 представлена диаграмма вариантов использования разрабатываемого информационного портала.

В данном проекте присутствуют 4 действующих лица: пользователь, не авторизованный пользователь, редактор и модератор.

Действующее лицо пользователь может просматривать страницу достопримечательности, оставлять комментарии и лайки, создавать группу для путешествия и вступать в уже существующую.

Действующее лицо не авторизованный пользователь может просматривать страницу достопримечательности.

Действующее лицо редактор может добавлять новые достопримечательности и изменять существующие.

Действующее лицо модератор может удалять комментарии и блокировать пользователя, если его слова или действия несут оскорбления или являются спамом.

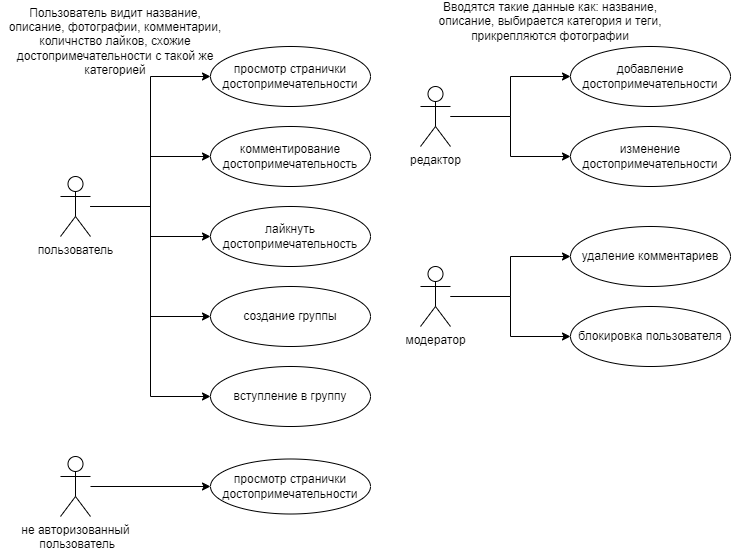


Рисунок 1.1 – UML диаграмма вариантов использования

* 1. **Обоснование выбора средств реализации**

Для разработки онлайн-гида необходимы:

* фронтенд веб-сайт;
* база данных;
* бэкенд веб-сайт.

Для разработки фронтенда веб-сайта используются язык разметки HTML, язык стилей CSS и язык программирования JavaScript [1]. Эти языки позволяют задавать внешний вид страницы и обеспечивать правильное отображение в популярных браузерах. Для работы с кодом можно использовать простой текстовый редактор, такой как Блокнот, или более удобные редакторы, например, VS Code или Sublime Text, которые облегчают работу с кодом, подсвечивают синтаксис и указывают на ошибки.

Перед началом верстки создается макет сайта, который помогает визуализировать будущий дизайн. Для этой цели хорошо подходит Figma, где можно создать черновой или более подробный макет, продумать расположение элементов интерфейса и выбрать цветовую гамму.

Данные о пользователях, материалах сайта, бронировании экскурсий и рассылках хранятся в базе данных, которую можно администрировать с помощью СУБД SQLite [2]. Эта система обладает не только простотой использования, но и встроенными механизмами безопасности, возможностью выполнения операций CRUD и использования SQL для запросов данных из JSON-документов, составления отчетов и статистики. Для визуального отображения таблиц и связей между ними можно использовать Draw.io, в котором доступен весь нужный функционал [3].

Для разработки бэкенда веб-сайта будет использоваться фреймворк Laravel. Laravel – бесплатный веб-фреймворк с открытым кодом, предназначенный для разработки с использованием архитектурной модели MVC

(англ. Model View Controller – модель-представление-контроллер). Laravel выпущен под лицензией MIT [4]. Для работы с этим фреймворком так же можно использовать VS Code.

**2** **Анализ данных**

Данные, с которыми работает веб-сервис можно разделить на три категории: входные, промежуточные и выходные данные.

Входные данные поступают на прямую от пользователя в программу. Промежуточные данные используются во время выполнения программы. Выходные данные выводятся пользователю.

Входными данными в данном проекте являются:

* учетные данные для входа;
* информация о пользователе: имя, фамилия, номер телефона, электронная почта, наличие машины, предпочтения в местах для путешествия;
* текст комментария;
* информация о создаваемой группе: место отправления, место прибытия, дата, вид перемещения.

Выходными данными в проекте являются:

* информационные страницы веб-приложения;
* комментарии о достопримечательности, оставленные другими пользователями;
* существующие группы, которые набирают людей для путешествия, в них отображаются: место отправления, место прибытия, дата, вид перемещения.

# 3 Программная реализация

Разрабатываемое веб-приложение разрабатывается на основе объектно-ориентированного подхода, поэтому оно состоит из классов и реализует стандарты такого подхода. Классы и связи между ними представлены в виде диаграммы классов на рисунке 3.1.

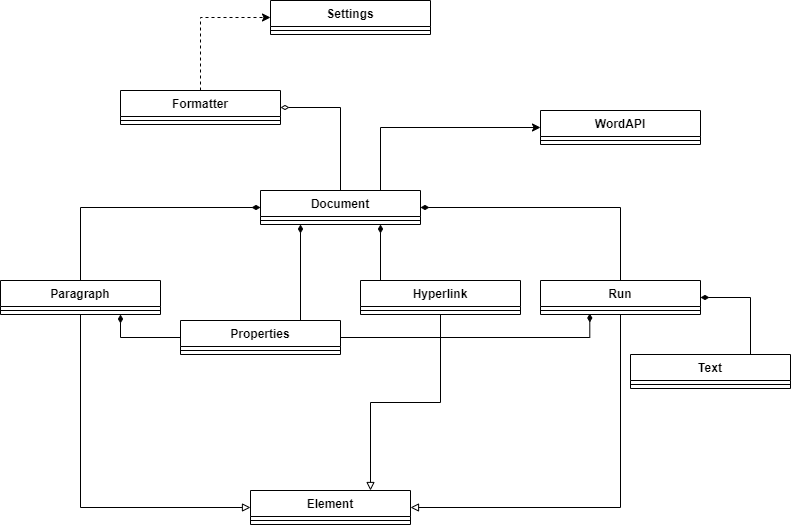


Рисунок 3.1 - Диаграмма классов

Для описания назначения классов, реализованных в веб-сервисе представлена информация в виде таблицы (таблица 3.1).

Таблица 3.1 - Описание классов

|  |  |
| --- | --- |
| Название класса | Назначение |
| Formatter | Класс для программы форматтер, которая форматирует электронные документы |
| Document | Класс для электронных документов |
| Hyperlink | Класс для ссылок в электронных документах |
| Paragraph | Класс для абзацев в электронных документах |
| Название класса | Назначение |
| Properties | Класс для настроек абзацев, вставок и т.д. |
| Run | Класс для пробегов по документу |
| Element | Абстрактный класс для элементов (Hyperlink, Run, Paragraph, Properties, Text) |
| WordAPI | Класс API для работы с документами Microsoft Word |
| Text | Класс для текстовых вставок |
| Settings | Класс для настроек документа |

Окончание таблицы 3.1

Структура создаваемого проекта представлена в виде диаграммы компонентов на рисунке 3.2.

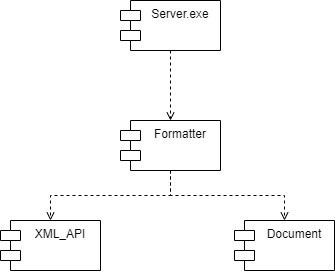


Рисунок 3.1 - Диаграмма компонентов

Для подробного описания компонентов представлена таблица 3.2.

Таблица 3.2 - Описание компонентов

|  |  |
| --- | --- |
| Компонент | Описание компонента |
| Server.exe | Компонент, представляющий основное приложение, запускающее серверную часть веб-сервиса |
| Formatter | Компонент, представляющий модуль, описывающий класс Formatter |
| XML\_API | Компонент, представляющий модуль, описывающий класс XML\_API |
| Document | Компонент, представляющий модуль, описывающий классы Document и Element, а также его потомки |

При разработке веб-редактора, который будет находится в браузере необходимо понимать, что возникнут сложности с преобразованием элементов WordXML в HTML. Далее будет описан алгоритм преобразования документа формата docx в документ, открытый в HTML.

Окончание таблицы 3.2

Документ типа docx состоит из нескольких файлов формата XML [6], чтобы их получить необходимо разархивировать такой файл. Основной файл, который будет интересовать во время работы это document.xml, который нам необходимо парсить и вычленить отдельные элементы. После этого элементы преобразуются в объекты класса Element, например, элемент w:p, который представляет собой абзац документа будет преобразован в объект класса Paragraph, являющийся классом-потомком класса-родителя Element.

Теперь, документ, представленный в виде множества объектов разных классов передаётся от сервера в скрипт, формирующий страницу на языке программирования JavaScript.

Основные данные, которые необходимо спарсить из документа docx это:

* тег элемента. Например, w:t, w:p, w:r и другие теги элементов WordXML. Список основных необходимых для данного приложения элементов представлен в таблице 3.2;
* стилевая настройка элемента. Обычно она содержится в элементе w:Pr. Каждая настройка реализована также в виде элемента XML. Список основных элементов стиля и их описание описано в таблице 3.2.

Таблица 3.3 - Основные элементы WordXML

|  |  |
| --- | --- |
| Название элемента | Описание элемента |
| w:t | Элемент текст. Содержит текст, который видит пользователь |
| w:p | Элемент параграф. Контейнер для текстовых вставок, ссылок, стиля параграфа |
| w:r | Элемент текстовая вставка. Контейнер для текста и стиля текста |
| w:hyperlink | Элемент ссылка. Контейнер для текстовых вставок и стиля ссылки, а также адреса, куда ссылка перенаправляет |
| Название элемента | Описание элемента |
| w:pPr | Элемент свойство параграфа. Содержит свойства параграфа, такие как выравнивание, отступ первой строки и другие |
|  |  |
| w:rPr | Элемент свойство текстовой вставки |
| w:body | Элемент тело документа. Контейнер, содержащий все элементы, описывающие документ |

Окончание таблицы 3.3

Таблица 3.4 - Основные элементы стилевой настройки WordXML

|  |  |
| --- | --- |
| Название элемента | Описание элемента |
| w:rFonts | Элемент шрифт. Указывает какой шрифт используется в текстовой вставке |
| w:ind | Элемент отступа. Указывает отступы в параграфе, например, отступ первой строки |
| w:sz | Элемент размер шрифта. Указывает размер шрифта в текстовой вставке |
| w:jc | Элемент выравнивание. Указывает выравнивание в параграфе |
| w:spacing | Элемент стиля, который указывает расстояние между линиями в параграфе |

Чтобы показать пользователю документ в веб-сервисе необходимо представить элементы в HTML-разметке и написать для них стили на языке стилей CSS. Для этого объекты класса Element и его потомков, сформированные после парсинга документа MS Word, передаются в скрипт, в виде JSON-строки. Из данной строки постепенно формируются элементы и вставляются в HTML-разметку.

Для удобства работы с документом в веб-редакторе было решено сделать объектно-ориентированную структуру документа, где параграфы и текстовые вставки представлены объектами, с которыми можно работать изолированно друг от друга. У каждого такого объекта есть редактируемые данные: текст элемента, стиль элемента а также его положение в документе. Чтобы изменить его текст, нужно нажать на нужный элемент и изменить его текст. Для изменения стилевой настройки предусмотрено дополнительное меню содержащее различные кнопки и другие элементы помогающие пользователю управлять объектами. Чтобы изменять положения в пространстве таких объектов в веб-сервисе реализована система Drag’n’Drop, позволяющая мышкой переносить элементы по документу.

Кроме этого в веб-сервисе есть возможность сохранить уже настроенные объекты для последующего копирования в других документах без повторного создания с нуля.

Также, для реализации основной функции формирования шаблонов документов были реализованы переменные документа, которые пользователь может создавать, называть и задавать значение. Смысл переменных в том, что пользователь может создать переменную, например ФИО, задать ей значение и расставить элементы, которые перенимают значение переменной ФИО, в документе, где надо, чтобы подставилось ФИО. Теперь шаблон готов, пользователю нужно только вбивать новое значение для переменной ФИО и документ готов для другого клиента.

# 4 Техническое задание

* 1. **Введение**

Современное общество становится все более осознанным в вопросах экологии и природы. Путешествия и отдых на природе становятся все популярнее, и многие люди стремятся открыть для себя уникальные природные достопримечательности своего края. Для этого необходимо иметь доступ к актуальной информации о таких местах и координировать поездки с другими интересующимися.

Информационный портал для природных достопримечательностей края предназначен для сбора, хранения и предоставления информации о природных объектах, а также для обеспечения возможности пользователей создавать и присоединяться к группам для организации поездок к этим местам.

# Назначение разработки

Целью данного проекта является создание веб-приложения, которое обеспечивает доступ к информации о природных достопримечательностях края, позволяет пользователям создавать аккаунты, присоединяться к группам, планировать поездки и делиться опытом с другими участниками. Главной задачей информационного портала является содействие в сохранении и популяризации природных ресурсов региона, а также в формировании сообщества людей, увлеченных и озабоченных экологическими вопросами.

# Требования к функциональным характеристикам

В программе предусмотрено 4 действующих лица.

Незарегистрированному пользователю представлены следующие функции:возможность зарегистрировать аккаунт, предоставив базовую информацию о себе;

1. просмотр списка природных достопримечательностей, включая описание, фотографии, геолокацию и отзывы других пользователей.

Зарегистрированному пользователю представлены следующие функции:

1. просмотр списка природных достопримечательностей, включая описание, фотографии, геолокацию и отзывы других пользователей;
2. возможность оставлять отзывы, комментарии и оценки к природным объектам и поездкам;
3. возможность создавать группы для посещения природных мест и приглашать других пользователей присоединиться к ним. Владельцы групп могут управлять участниками, планировать поездки и обмениваться сообщениями внутри группы. Должна быть возможность установить дату, время и место встречи для поездки;
4. с помощью системы обратной связи с администраторами портала пользователь может сообщить об изменениях в статусе природных объектов;
5. возможность оставить жалобу на другого пользователя. Модератору представлены следующие функции:
   1. отслеживание размещения не нежелательной информации или спама;
   2. удаление отзывов пользователей, если отзыв не соответствует правилам размещения;
   3. временная или постоянная блокировка пользователя за частое нарушение правил размещения.

Редактору представлены следующие функции:

1. добавление, изменение и удаление тэгов;
2. добавление, изменение и удаление категорий;
3. добавление, изменение и удаление публикаций. Администратору представлены следующие функции:
   1. доступ к административной панели для управления пользователями,

публикациями и другими аспектами системы;

* 1. возможность собирать статистическую информацию о популярности природных объектов, активности пользователей и другие аналитические данные.

# Требования к составу и параметрам техническим средствам

Минимальные технические требования:

1. процессор: Intel Core i3 или аналогичный AMD;
2. оперативная память: 4 Гб ОЗУ;
3. место на диске: 500 MB. Оптимальные технические требования:
   1. процессор: Intel Core i7 или аналогичный AMD с тактовой частотой не менее 3 ГГц;
   2. оперативная память: 8 Гб ОЗУ;
   3. место на диске: 1ГБ.

# Требования к информационной и программной совместимости

Программа должна функционировать под управление ОС семейства Windows не ниже версии 7.

# Требования к программной документации

Программная документация должна содержать руководство пользователя.

# 5 Руководство пользователя

Чтобы начать работу в веб-сервисе необходимо создать пустой документ, либо загрузить уже готовый документ. Для этого нужно нажать на одну из кнопок «Создать документ» или «Загрузить новый документ». На рисунке 5.1 показаны данные кнопки.

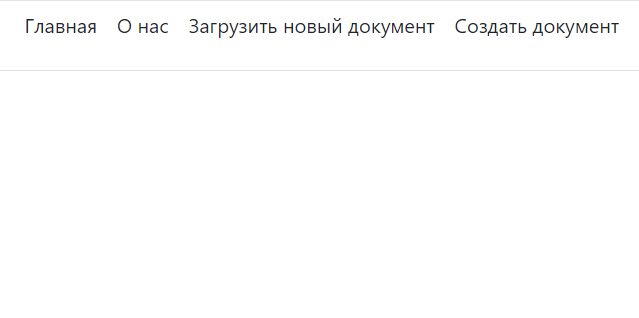


Рисунок 5.1 - Кнопки, необходимые для начала работы с документами

Чтобы создать документ нужно ввести его название, с которым он будет хранится в базе данных веб-сервиса, описание и автора документа. Если необходимо загрузить готовый документ, то дополнительно нужно выбрать файл на компьютере, который вы хотите загрузить. Форма для заполнения представлена на рисунке 5.2.

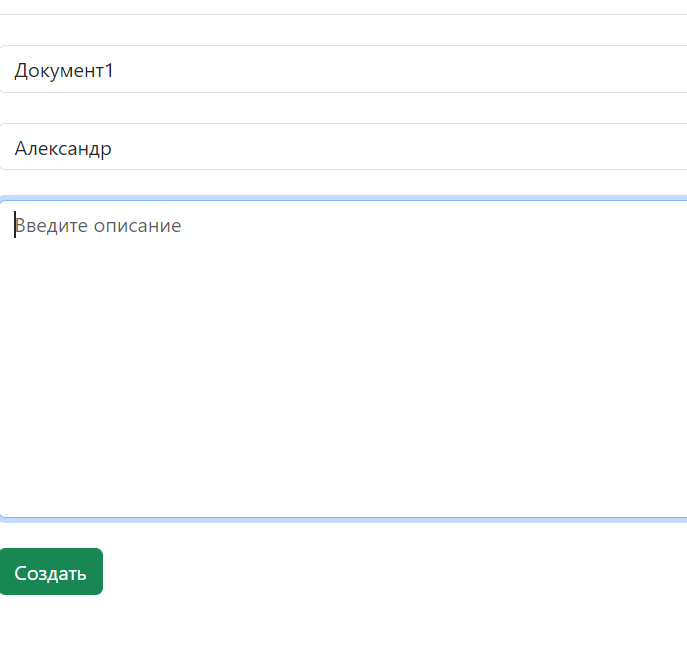


Рисунок 5.2 - Окно создания документа

После создания документа он открывается в веб-редакторе и становится возможным управление текстовым документом. Веб-редактор представлен на рисунке 5.3.

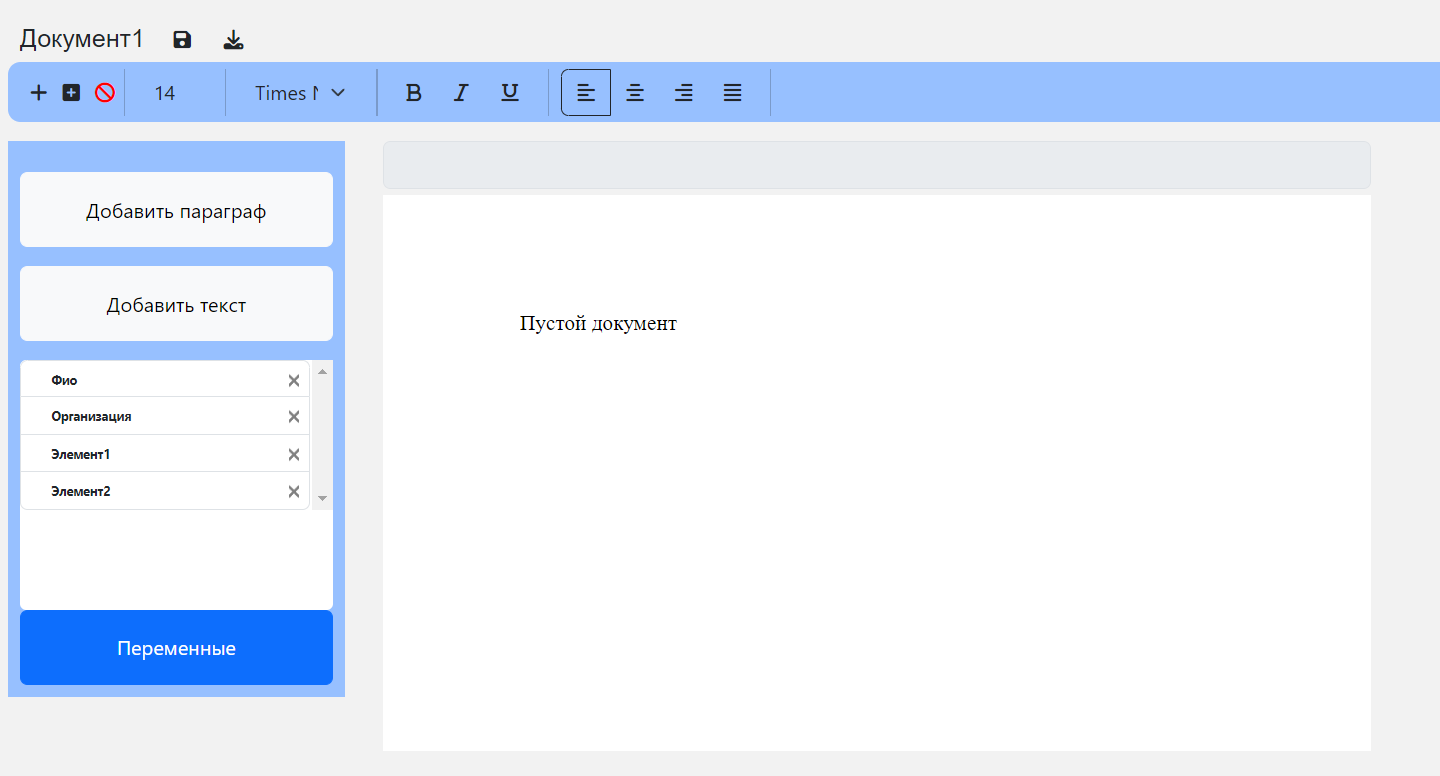


Рисунок 5.2 - Веб-редактор текстовых документов

В документе всё представлено двумя видами объектов: параграф и текстовая вставка. Их можно переносить мышкой при помощи Drag’n’Drop. Объект параграф и текстовая вставка на рисунке 5.4.

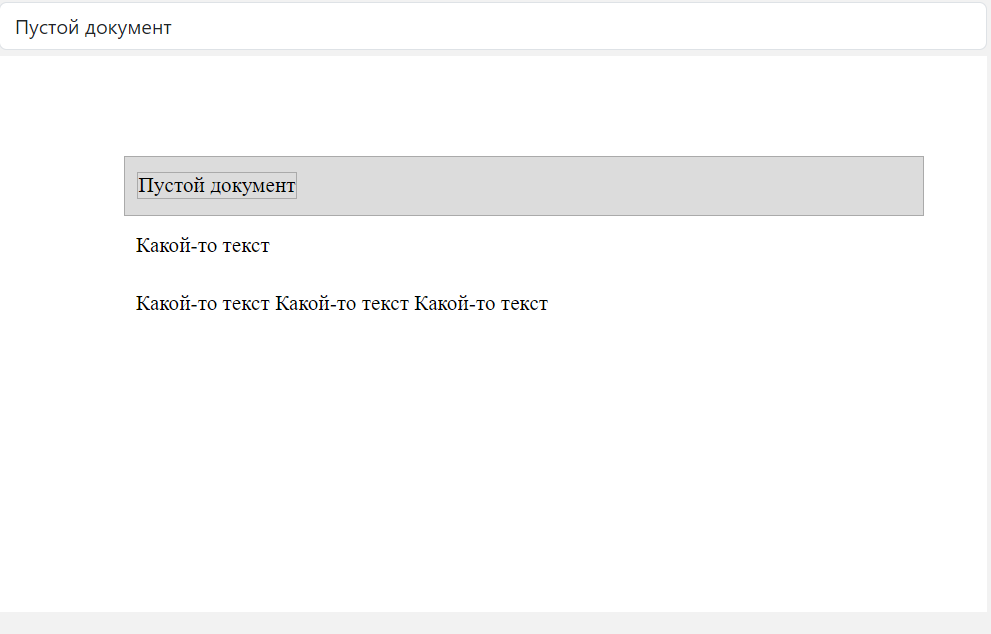


Рисунок 5.3 - Объект параграф и текстовая вставка

Чтобы перенести объект в другое место нужно зажать мышкой по интересующему объекту и начать переносить мышкой. Элемент, который переносится будет подсвечен при помощи зелёной рамки до конца переноса, чтобы пользователь не потерял объект во время переноса. Процесс переноса элемента изображен на рисунке 5.5.



Рисунок 4.5 - Процесс переноса элемента в документе

Элементам можно задавать стилевую настройку при помощи меню стилевых настроек. На этом меню есть возможность изменить шрифт, размер шрифта, сделать текст жирным, курсивом и подчеркнуть. Если элемент параграф, то можно настроить его выравнивание, отступ первой строки, а также настроить все элементы параграфа выделив только его. Меню стилевых настроек изображено на рисунке 5.6. Результат таких настроек можно увидеть на рисунке 5.7.

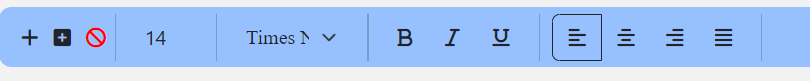


Рисунок 5.5 - Меню стилевых настроек.

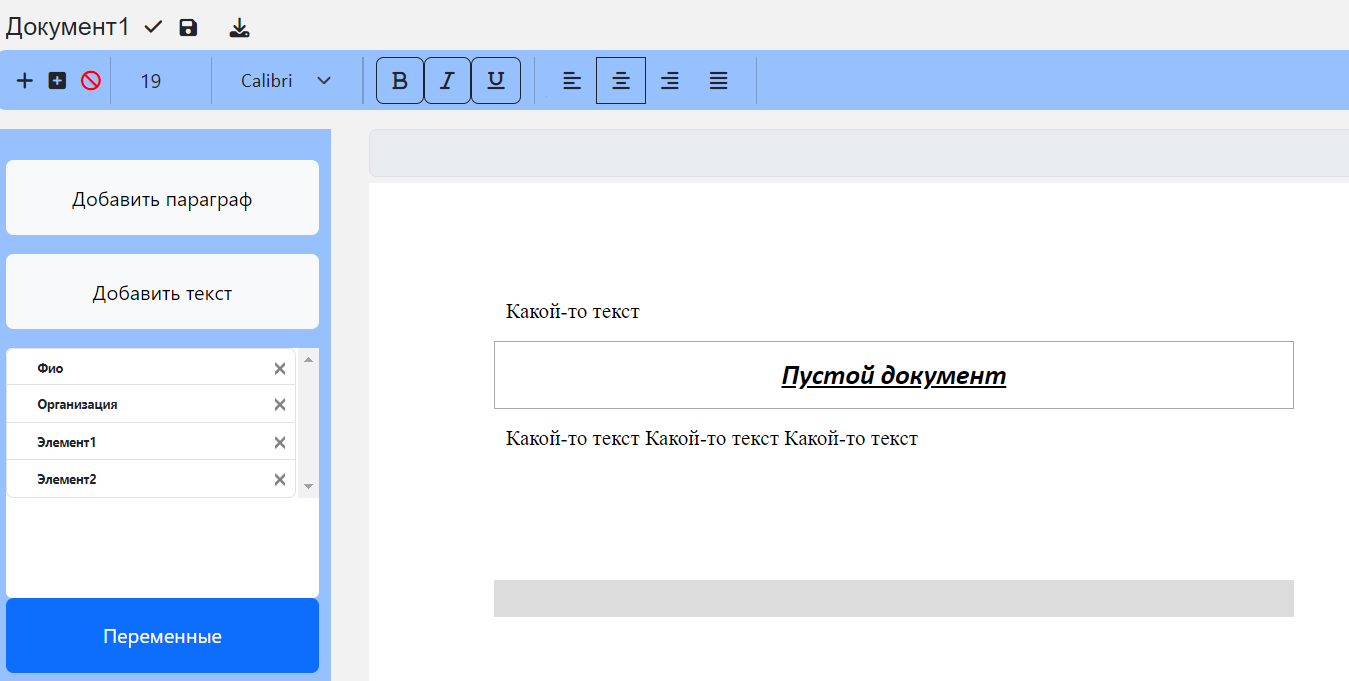


Рисунок 5.7 - Результат применения стилевых настроек.

Для изменения текста в текстовых вставках можно воспользоваться двумя способами:

1. Нажать два раза на интересующий текстовый элемент и редактировать его. Данный метод представлен на рисунке 5.8.
2. Использовать поле редактирования текста, находящееся выше документа. Этот метод представлен на рисунке 5.9.

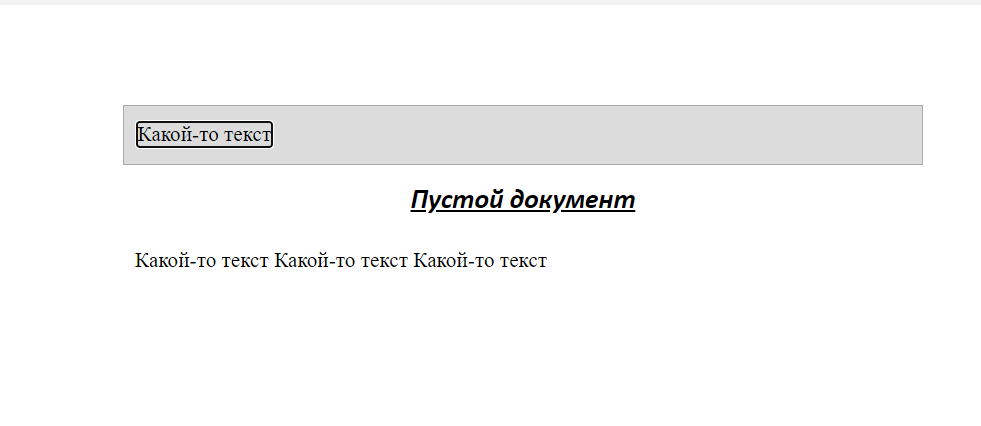


Рисунок 5.8 - Редактирование текстового элемента в документе через двойной клик

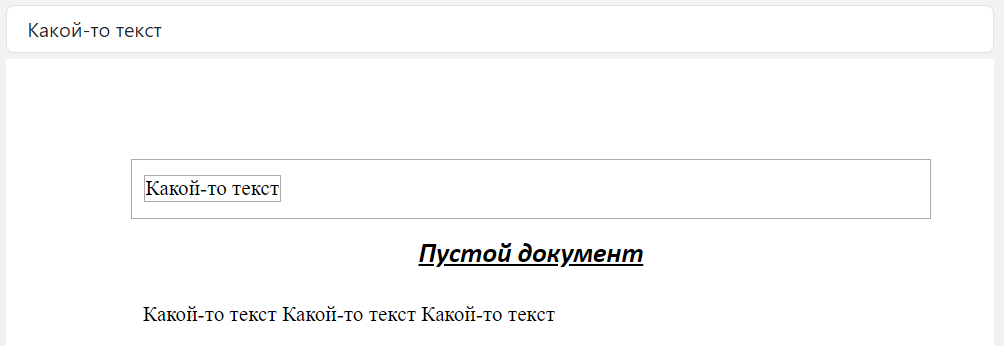


Рисунок 5.9 - Поле редактирования текста

Чтобы добавить новые элементы на документ необходимо нажать на соответствующие кнопки. Кнопки находятся на меню стилевой настройки и представлены иконками. Добавление параграфа – плюс без фона, добавление текстовой вставки – плюс на черном фоне. Данные кнопки изображены на рисунке 5.10. Если нужно добавить параграф в конец документа, то можно просто нажать на кнопку добавления параграфа, в случае, если нужно добавить параграф после какого-то, то нужно выбрать параграф, после которого будет вставлен новый и нажать на кнопку. Чтобы добавить текстовую вставку, нужно выделить параграф, в который хотите добавить данный элемент и нажать на кнопку добавления текстовой вставки. Выбор куда вставить элемент такой же как и с параграфом.

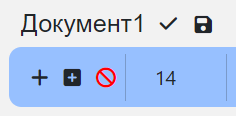


Рисунок 5.10 - Кнопки добавления новых элементов

Для сохранения изменений в документе, нужно нажать на кнопку сохранения документа, которая находится выше меню стилевой настройки рядом с названием документа. Если сохранение прошло успешно, то рядом появляется галочка, оповещающая пользователя о том, что документ успешно сохранён. Также рядом есть кнопка загрузки текущего документа на компьютер пользователя. Данные служебные кнопки изображены на рисунке 5.11.



Рисунок 5.11 - Служебные кнопки документа в веб-редакторе.

На веб-сервисе есть возможность сохранить любой элемент для дальнейшего использования в этом же документе, либо в других уменьшая нагрузку на пользователя тем, что ему не придётся заново создавать такой элемент. Меню сохранения элементов находится на левом боковом меню. Меню сохранения изображено на рисунке 5.12.

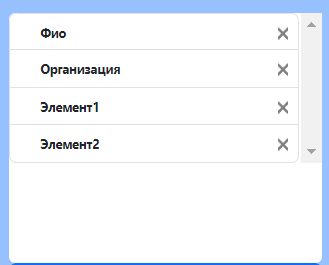


Рисунок 5.12 - Меню сохранения элементов

Чтобы сохранить элемент пользователь должен перенести мышкой такой элемент в меню сохранения. Когда меню будет готово сохранить элемент оно подсветится зелёной рамкой (см. рис. 5.13). Элемент всегда добавляется в конец списка. Его можно удалить из списка при помощи кнопки в виде крестика.

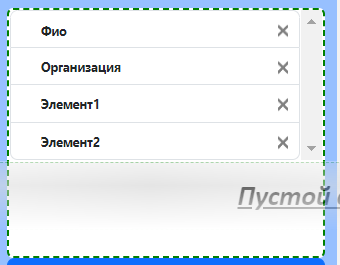


Рисунок 5.13 - Процесс сохранения элемента

Элемент в меню сохранения можно переименовать нажав на название и изменив его. Чтобы добавить сохранённый элемент в документ пользователю нужно перенести мышкой из меню сохранения на документ. Веб-редактор сам понимает является ли элемент параграфом или текстовой вставкой и будет предлагать поставить только в нужные места.

Веб-сервис предлагает функция создания переменных в текстовых документах. Они позволяют уменьшить концентрацию пользователя на заполнении в документе повторяющихся текстовых величин, например, ФИО клиента. Чтобы создать переменную необходимо нажать на кнопку «Переменные» в левом боковом меню (см. рис. 5.14).

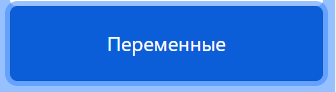


Рисунок 5.14 - Кнопка для открытия меню работы с переменными документа

Меню работы с переменными документа представляет собой таблицу и кнопки для добавления и удаления переменных в таблице. Это меню можно увидеть на рисунке 5.15.

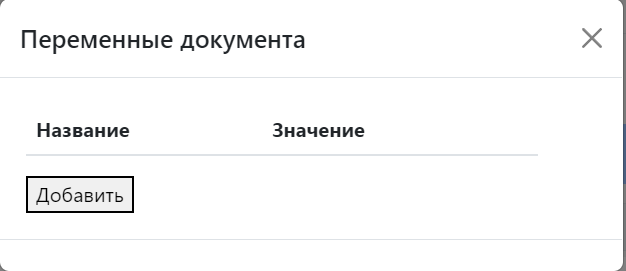


Рисунок 5.15 - Меню для работы с переменными документа

После добавления переменной в таблице ему можно задать название и значение для этой переменной. Готовая переменная представлена на рисунке 5.16.

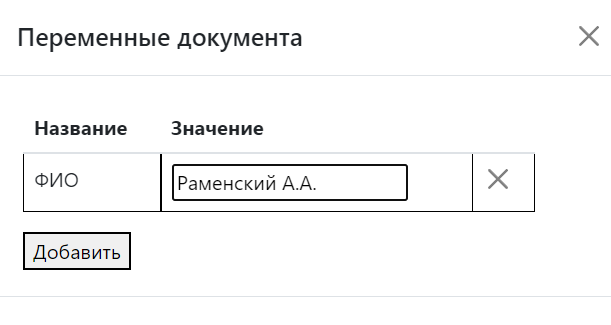


Рисунок 5.16 - Готовая переменная для документа

Чтобы использовать переменную в документе нужно вписать в поле для изменения текста при выборе текстовой вставки название переменной в специальном формате: {: <имя переменной> :}. После этого текст в документе изменится на значение переменной, а в поле для изменения текста будет написано имя переменной. Применение переменной в тексте показано на рисунке 5.17. Переменных в документе может быть несколько, а также одна переменная может применяться в документе не один раз.

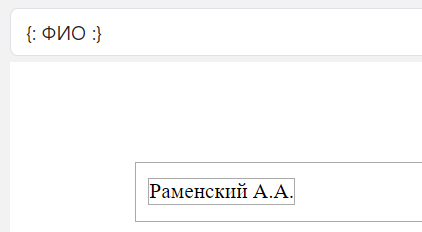


Рисунок 5.16 - Применение переменной в документе

# ЗАКЛЮЧЕНИЕ

В ходе проделанной работы был разработан веб-сервис для разработки шаблонов и автоматизации стилевой настройки документов. Создан веб-редактор документов, в котором пользователь может изменить стиль и текст документа, а также расставить переменные и заполнить их.

Разработанное программное средство выполняет те функции, которые были запланированы. Пользователь может создавать типовые документы и использовать их в будущем для своих клиентов, заполняя только переменные документа и почти не трогая сам текст.

Система масштабируема и будет расширяться добавляя больше функций для более удобной работы с документами. Планируется подключение переменных к базе данных организации, чтобы они автоматически подставлялись в документе на нужные места.

# СПИСОК ИСПОЛЬЗУЕМЫХ ИСТОЧНИКОВ

1. Понятие электронного документа. – Текст: электронный // Информационная система 1С:ИТС. – 2017. – URL: https://its.1c.ru/db/eldocs/content/12/hdoc (дата обращения 01.03.2023).
2. Шаблон электронного документа. – Текст: электронный // Документация системы Галактика ERP (Версия 8.10). – 2009. – URL: https://www.erpandcrm.ru/ru/directum\_unitedhelp/extfile/directum/2954653\_shabl ony\_jelektronnyh\_dokumentov\_opisani.htm#:~:text=Шаблон%20электронных%20документов%20тип%20карточки,запись%20в%20справочнике%20создается%20автоматически (дата обращения 01.03.2023).
3. Введение в JavaScript. – Текст: электронный // Современный учебник JavaScript. – 2022. – URL: https://learn.javascript.ru/intro/ (дата обращения 03.03.2023).
4. Bootstrap: вступление. – Текст: электронный // Bootstrap. − 2019. − URL: https://getbootstrap.ru/docs/5.1/getting-started/introduction/ (дата обращения 07.03.2023).
5. Эффективный Django. Часть 1. – Текст: электронный // Хабр – 2014. − URL: https://habr.com/ru/post/240463/ (дата обращения 12.03.2023).
6. Обзор Microsoft Office Word. – Текст: электронный // Исходники.Ру программирование – 2007. – URL: http://www.ishodniki.ru/art/art\_progr/ net/655.html (дата обращения 01.03.2023).
7. Структура документа WordprocessingML. – Текст: электронный // Microsoft. – 2022. – URL: https://learn.microsoft.com/ru-ru/office/open-xml/structure-of-a-wordprocessingml-document (дата обращения 01.03.2023).