Министерство образования и науки Республики Казахстан

Северо-Казахстанский университет им. М. Козыбаева

СЕМЕСТРОВЫЙ ПРОЕКТ 3

На тему: «Исследование и разработка программы анализа текста»

ИКТ1АПО.02 СП

|  |  |
| --- | --- |
| Выполнили студенты  группы АПО-19 | Бурханов Р.Р.  Вагина О.А.  Семейников А.Н. |
| Научный руководитель  доцент, PhD | Астапенко Н.В. |

Петропавловск, 2022

СОДЕРЖАНИЕ

[ВВЕДЕНИЕ 3](#_Toc98280509)

[1 Аналитическая часть 5](#_Toc98280510)

[1.1 Исследование предметной области 5](#_Toc98280511)

[2 Проектирование автоматизированной информационной системы 6](#_Toc98280512)

[2.1 Распределение задач 6](#_Toc98280513)

[2.2 Описание баз данных 7](#_Toc98280514)

[2.3 Разработка архитектуры программного средства 10](#_Toc98280515)

[2.4 Описание используемых алгоритмов 12](#_Toc98280516)

[2.5 Тестирование программного средства 17](#_Toc98280517)

[ЗАКЛЮЧЕНИЕ 21](#_Toc98280518)

[СПИСОК ИСПОЛЬЗУЕМОЙ ЛИТЕРАТУРЫ 22](#_Toc98280519)

# ВВЕДЕНИЕ

Современное развитие информационных технологий и глобальной сети Интернет предоставило широким кругам пользователей доступ к огромным массивам информации. Появилось большое число online-библиотек, содержащих художественную и научно-техническую литературу. Стало возможным читать книги, новости и газеты непосредственно с экрана компьютера.

В сети Интернет стало доступно множество методических указаний, курсов лекций, учебников и т.д. Кроме того, появились огромные коллекции рефератов, готовых лабораторных работ, курсовых и дипломных проектов и даже диссертаций. Использование компьютерной техники сильно облегчило задачу поиска и копирования подобной информации. Если раньше для написания реферата или контрольной работы информацию было нужно, по крайней мере, найти в книгах и переписать (вручную, перепечатать или ввести в компьютер с помощью сканера и программ распознавания текстов), то теперь достаточно ввести название темы в поисковую систему и скопировать найденные материалы. Стал распространяться метод написания работ, получивший название «Copy & Paste». Метод заключается в простом копировании информации из одного или нескольких источников с минимальным редактированием получающегося таким образом текста.

Аналогичная ситуация наблюдается с отчетными материалами внутри учебных заведений. В связи с тем, что большое число пояснительных записок по курсовым и дипломным проектам выполняется с использованием компьютеров, происходит их распространение и повторное использование среди учащихся.

В последнее время наблюдается бурный рост использования в учебном процессе подобной заимствованной информации. Ситуация усугубляется тем, что учащиеся иногда не знают (не читают) то, что написано и «их» работах. Плагиат – умышленное присвоение авторства чужого произведения науки или искусства, чужих идей или изобретений. Как можно убедиться из определения, подобные заимствованные работы можно отнести к разряду плагиата. Задача обнаружения недобросовестного использования заимствованных текстов в учебных и ученых кругах (фактов плагиата) приобретает высокую актуальность.

В статье «Плагиат в работах студентов и аспирантов: проблема и методы противодействия», авторами которой являются Никитов А.В., Орчаков О.А., Чехович Ю.В. указано «с появлением Интернета письменные работы студентов стали демонстрировать не степень развитости интеллектуальных и профессиональных навыков, а уровень умения студента «отловить» текст в Интернете и преподнести его как свою оригинальную работу. Причем этот процесс охватил весь спектр студенческих работ – от маленьких эссе до дипломных проектов и диссертаций. Еще более обостряет проблему то, что современные студенты владеют информационными технологиями лучше и пользуются ими чаще, чем российские вузовские преподаватели. Пожилой профессор, читая принесенную ему работу, восхищается эрудицией студента, стилем и аргументацией, удачным употреблением профессиональной терминологии и т.д. не подозревая, что этот текст написал и опубликовал в Сети его коллега из другого университета.

Побуждение студента самостоятельно написать различные работы, отчеты и многое другое не всегда имеет положительные результаты. Привычные методы воспитания – убеждение, внушение, проработка на собраниях и т. д. – здесь не действуют. Формирование ментальности – процесс длительный и относительно болезненный, и в ближайшей перспективе надеяться на то, что студенты перестанут списывать из Интернета, наивно и бесполезно. Нужно создавать стимулы, чтобы студентам было невыгодно заниматься плагиатом». В данном высказывание показана *актуальность* семестрового проекта. Ответом на данный вопрос являются системы проверки текста на плагиат, разрабатываемая в данном семестровом проекте. Данная система анализа текста позволит определять текст на заимствование из других источников, что поможет бороться с плагиатом в работах студентов.

*Целью* семестрового проекта является исследование и разработка программы анализатора эссе, обеспечивающая исправление текст по «.,», проверку количества слов эссе на соответствие требованию, проверка количества слов в предложении (<24), анализ тошноты текста. Преподаватель имеет возможность устанавливать лимит на количество слов в эссе, сохраняет одно/все эссе в отдельные word-файлы, проверяет эссе на плагиат. Студент имеет возможность загружать эссе.

Для осуществления обозначенной цели необходимо выполнить следующие *задачи*:

* анализ предметной области;
* выбор средств проектирования программного средства и баз данных;
* проектирование и создание базы данных;
* разработка архитектуры программного средства;
* реализация алгоритмов программного средства;
* отладка и тестирование программного средства.

*Практическая значимость* семестровой работы заключается в использовании результатов работы в практической деятельности образовательных учреждений, систем автоматической обработки текста для анализа русскоязычного текста. Разработанные в ходе семестрового проекта алгоритмы и программное обеспечение может использоваться как в качестве самостоятельного продукта позволяющего облегчить анализ информации в больших объемах текстовых данных, так и в качестве основы для разработки узкоспециализированной системы семантической проверки текста.

# 1 Аналитическая часть

## 1.1 Исследование предметной области

Предметной областью семестрового проекта является исследование способов анализа текста. Для анализа текста студенту необходимо ввести выполненное задание в текстовом формате. Текст перед отправкой преподавателю: исправляется по «.,», проверяется количество слов эссе на соответствие требованию и количество слов в предложении (<24), анализируется тошнота текста. Функции допустимые преподавателю в разрабатываемом сайте: устанавливает лимит на количество слов в эссе, сохраняет одно/все эссе в отдельные word-файлы, проверяет эссе на плагиат. Таким образом сайт должна содержать несколько ролей входа это студент и преподаватель, поскольку функции у каждого из них различны.

В основе поиска совпадений по системе антиплагиата лежит семантический анализ текста. То есть, первоначальное разбиение его на отдельные части, дробление их на составляющие отрезки, а затем уже программа осуществляет по каждому отрезку поиск совпадений в сети интернете, отслеживание ссылок на соответствующие документы, подсвечивание их для автора и выдачу полного отчета по обработанному отрывку. Антиплагиат предоставляет информацию об общем уровне уникальности документа в процентах. При нахождении схожих или идентичных отрывков текста в сети, программа антиплагиат помечает эти отрывки цветным маркером, указывая после общего анализа ссылки на источники плагиата. Кроме подсветки не уникальных фрагментов, антиплагиат предоставляет информацию об общем уровне уникальности документа в процентах. Она высчитывается по установленному алгоритму. Абсолютная уникальность текста по антиплагиату – 100 % это величина достаточно условная. В любом случае, рассчитанная величина не является абсолютной истиной, она всего лишь свидетельствует об уровне владения языком автором текста, его умением обрабатывать и оригинально предоставлять информацию и не использовать чужие тексты в своих работах в первоначальном виде.

Помимо общего уровня антиплагиата система указывает на ошибки в тексте, исправляет текст по возможности. Сравнивает длину эссе с данными, которые ранее были введены преподавателем и при несоответствии требованиям запрещает отправку текста до тех пор, пока они не выполнены.

Таким образом, во время первого спринта было выявлено, что для работы автоматизированной информационной системы необходимо разработать следующие алгоритмы: исправление текста по «.,», проверка эссе на соответствие требованиям, анализ текста на тошноту, проверка на плагиат. После завершения исследования предметной области следует приступить к проектированию сайта проверки на антиплагиат.

# 2 Проектирование сайта анализа текста

## 2.1 Распределение задач

Управление ресурсами – это процесс планирования ресурсов и составления графика работ, выполняемых командой. Ресурсом может быть что угодно – от оборудования и финансовых средств до программного обеспечения и труда ваших сотрудников – всё, что помогает выполнить проект. В современном мире, где организации работают быстро, причём зачастую в виртуальной среде, управлять загруженностью сотрудников не так просто. Разработка плана управления ресурсами позволяет держать под контролем загрузку отдельных сотрудников и группы в целом в реальном времени. Благодаря этому возможно эффективно балансировать ресурсы и иметь цельную картину того, как коллектив будет достигать поставленные цели.

В методологии Scrum используются итерации фиксированной длины, называемые спринтами. В семестровом проекте спринт занимает 1 неделю. Scrum подразумевает три роли: владелец продукта, Scrum-мастер и участники команды разработчиков. В роли Scrum-мастера выступает капитан семестрового проекта Семейников Артем, в роли команды разработчиков и владельца продукта – группа студентов.

Группа, участвующая в семестровом проекте на тему: «Разработка сайта анализа эссе», состоит из 3 студентов. Техническое задание представляет собой создание макетов проекта, исследование предметной области, проектирование и подключение баз данных, создание алгоритмов, архитектуры проекта и тестирование итогового ПО. Распределение технического задания представлено в таблице 1.

Таблица 1. Распределение обязанностей технического задания

|  |  |  |  |
| --- | --- | --- | --- |
| Студент  Неделя | Семейников Артем | Вагина Оксана | Бурханов Руслан |
| 10 неделя | Выбор методологии, распределения обязанностей | Загрузка проекта в GitHub, изучение методологии | Исследование предметной области |
| 11 неделя | Создание макета базы данных | Создание макета сайта | Создание архитектуры ПО |
| 12 неделя | Создание и подключение базы данных | Верстка макета | Реализация алгоритма ограничения длины |
| 13 неделя | Реализация алгоритма проверки уникальности | Реализация алгоритма проверки тошноты текста | Реализация алгоритма изменения по «.,» |
| 14 неделя | Тестирование программы | Тестирование программы | Тестирование программы |

Составление отчета включает в себя написание введения, аналитической части, проектной части, заключения, а также добавления списка использованной литературы. Аналитическая часть включает в себя исследование предметной области, проектная часть – описание баз данных, архитектуры и алгоритмов проекта, тестирование программного средства. Распределение обязанностей по выполнению отчета представлено в таблице 2.

Таблица 2. Распределение обязанностей выполнения отчета

|  |  |  |  |
| --- | --- | --- | --- |
| Студент  Неделя | Семейников Артем | Вагина Оксана | Бурханов Руслан |
| 10 неделя | Распределение обязанностей | Оформление введения, описание выбранной методологии | Оформление исследования предметной области |
| 11 неделя | Добавление описание базы данных в отчет | Добавление диаграммы архитектуры | Добавление архитектуры в отчет |
| 12 неделя | Добавление диаграмм базы данных в отчет | Добавление макета сайта в тестирование проекта | Добавление алгоритма ограничения длины в отчет |
| 13 неделя | Добавление алгоритма уникальности в отчет | Добавление алгоритма проверки на тошноту в отчет | Добавление алгоритма изменения по «.,» |
| 14 неделя | Добавление тестирования | Форматирование отчета, добавление заключения и списка используемой литературы | Добавление тестирования |

После распределения работ можно приступить к выполнению задания в следующем порядке: первым этап является проектирование баз данных. Спроектированные базы данных в дальнейшем должны быть описаны в архитектуре программного средства. Следующим этапом является разработка общих алгоритмов проекта, а также алгоритмов обработки данных о пользователях и товарах. После подключения баз данных и сборки алгоритмов и архитектуры проекта, проводится тестирование программного средства.

## 2.2 Описание баз данных

В настоящее время жизнь человека зависит от различного рода информации, для управления которой требуются создания огромного количества баз и банков данных различного назначения. Изучение и проектирование баз данных относится ко второму спринту в соответствии с распределением задач.

Понятие базы данных (БД) можно применять к любой связанной между по определенному признаку информации, хранимой и ограниченной особым образом- что выполняется в СУБД SQL в виде таблиц. По сути БД – это некоторое подобие картотеки, электронного хранилища данных, которые хранятся в компьютере в виде одного или нескольких файлов. Данный семестровый проект предполагает создание базы данных сайта.

Инфологическая модель БД – это модель, которая описывает данные предметной области с использованием естественного языка.

Проектирование инфологической модели является основной задачей при создании БД. Цель инфологической модели- обеспечение наиболее естественных для человека способов сбора и представления той или иной информации, которую предполагается хранить в создаваемой базе. Поэтому инфологическую модель данных пытаются строить по аналогии с естественным языком (последний не может быть использован в чистом виде из-за сложности компьютерной обработки текстов и неоднозначности любого естественного языка). Основными конструктивными элементами инфологических моделей являются сущности, связи между ними и их свойства.

Хранение информации происходит в структурированном виде в виде таблиц. Первой таблицей является таблица «users». Данная таблица хранит в себе данные о логине и пароле пользователя, а также статус входа. Статус входа подразумевает две роли: преподаватель и студент. Информация будет представлена по следующим пунктам: код пользователя, логин, пароль, статус. При помощи данной информации происходит определение пользователя и вход в систему антиплагиата.

Второй таблицей является таблица «essay». Данная таблица хранит данные об эссе и сами эссе в базе данных. Информация, которая представлена в таблице это идентификатор эссе, идентификатор пользователя, идентификатор задания, текст и информацию о проверке на уникальность и тошноту.

Третьей таблицей является таблица «task». Данная таблица содержит идентификатор пользователя, который создал это задание, идентификатор задания, тема, ограничение по количеству слов, необходимая уникальность.

Инфологическая модель БД семестрового проекта представлена в соответствии с рисунком 2.1.

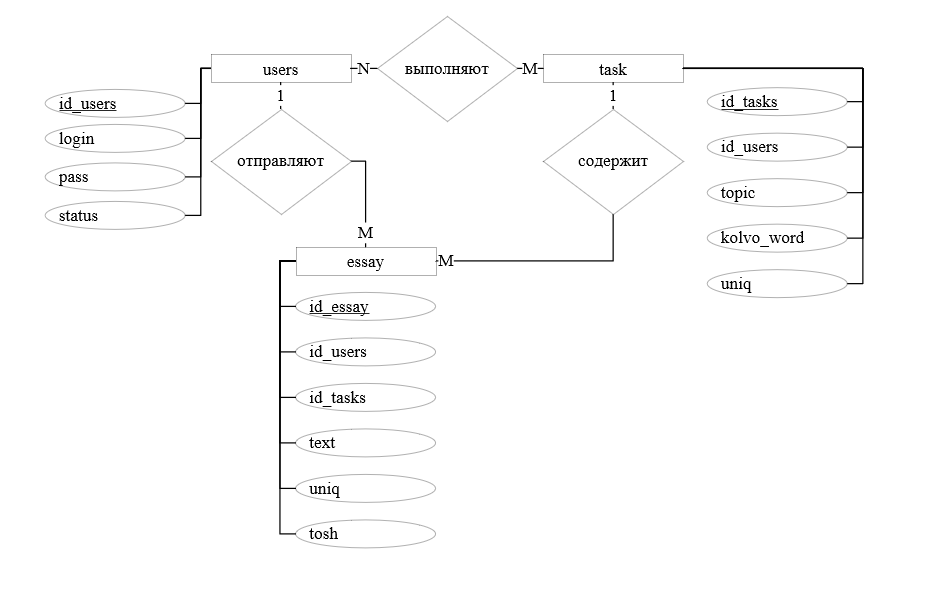


Рисунок 2.1 Инфологическая модель БД

После построения инфологической модели можно приступать к построению даталогической модели.

Даталогическая (Концептуальная) модель БД – это логическая организация данных и их взаимосвязь. Структурирование данных выполняется в соответствии с выбранной СУБД, которая строится на основе выбранной модели представления данных: иерархической, сетевой или реляционной. На данном этапе проектировщик создает структуру данных и организует связь между объектами.

Проектирование даталогической модели- важный этап в проектировании БД. На этом этапе важно правильно выделить сущности и описать их атрибуты. Ошибка на этом этапе может обернуться разработчику значительными потерями времени и сил в дальнейшем.

Даталогическая (концептуальная) модель- модель описывающая логику организации данных. Датологическое проектирование заключается в проектировании логической структуры БД. Таким образом, главное отличие даталогической модели от инфологической состоит в том, что инфологическая модель хранит в себе всю информацию о предметной области, необходимую и достаточную для проектирования базы данных, но она не привязана к определенной СУБД.

Даталогическое проектирование сводится к следующим этапам:

1. Определение таблиц.
2. Определение полей таблиц.
3. Определение типов данных в соответствии с выбранной СУБД.
4. Определение длины каждого поля таблиц.
5. Определение обязательности каждого поля.
6. Определение индексации каждого поля.

Даталогическая модель БД представлена в соответствии с рисунком 2.2

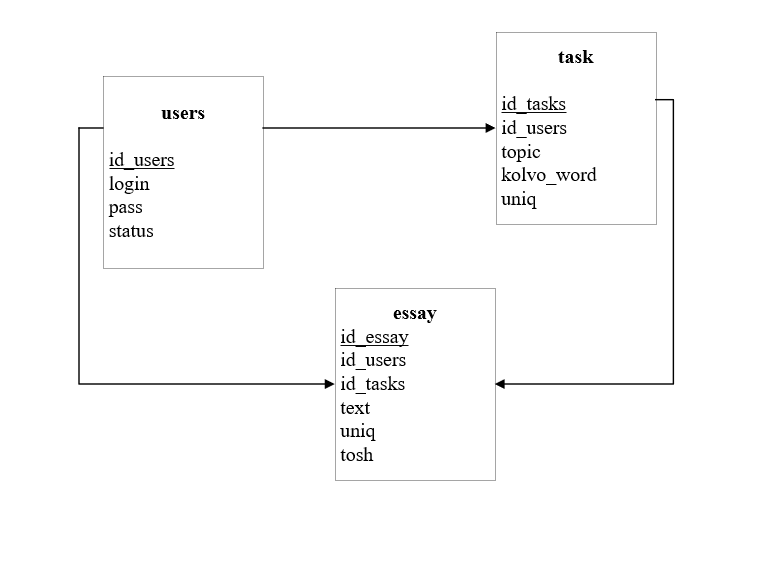


Рисунок 2.2 Даталогическая модель БД

Структурирование данных выполняются на основе модели реляционного представления данных.

Реляционная модель (от лат. relation- отношение) представляет собой совокупность простейших двумерных таблиц, связь между которыми осуществляется через одинаковые ключевые поля. Каждая строка такой таблицы называется записью, а столбец- полем. Каждая таблица описывает сущность предметной области. Результатами второго спринта являются инфологическая и даталогическая модели БД. Помимо проектирования баз данных второй спринт содержит разработку архитектуры программного средства.

## 2.3 Разработка архитектуры программного средства

Архитектура программного средства (ПС) – это его строение как оно видно (или должно быть видно) извне его, т.е. представление ПС как системы, состоящей из некоторой совокупности взаимодействующих подсистем. В качестве таких подсистем выступают отдельные файлы. Разработка архитектуры является первым этапом борьбы со сложностью ПС, на котором реализуется принцип выделения относительно независимых компонентов.

Система позволяет студенту вводить текст эссе. Система перед отправкой преподавателю: исправляет текст по «.,», проверяет количество слов эссе на соответствие требованию, проверяет количество слов в предложении (<24), анализирует тошноту текста. Функции допустимые преподавателю в системе анализа текста: устанавливает лимит на количество слов в эссе, сохраняет одно/все эссе в отдельные word-файлы, проверяет эссе на плагиат. Таким образом система должна содержать несколько ролей входа это студент и преподаватель, поскольку функции у каждого из них различны.

Система объединяет в себе файлы страницы со стороны студента, страницу авторизации и страницу со стороны преподавателя. В свою очередь Страницы со стороны преподавателя и со стороны студента делятся на функциональные блоки. Страница студента подразумевает блоки: просмотр задания, проверка текста, отправка задания. Страница преподавателя в свою очередь имеет блоки размещения задания и проверка задания. С БД связаны страница авторизация, и блоки отправка задания и размещение задания. Архитектура, разработанная во втором спринте представлена в соответствии с рисунком 2.3.

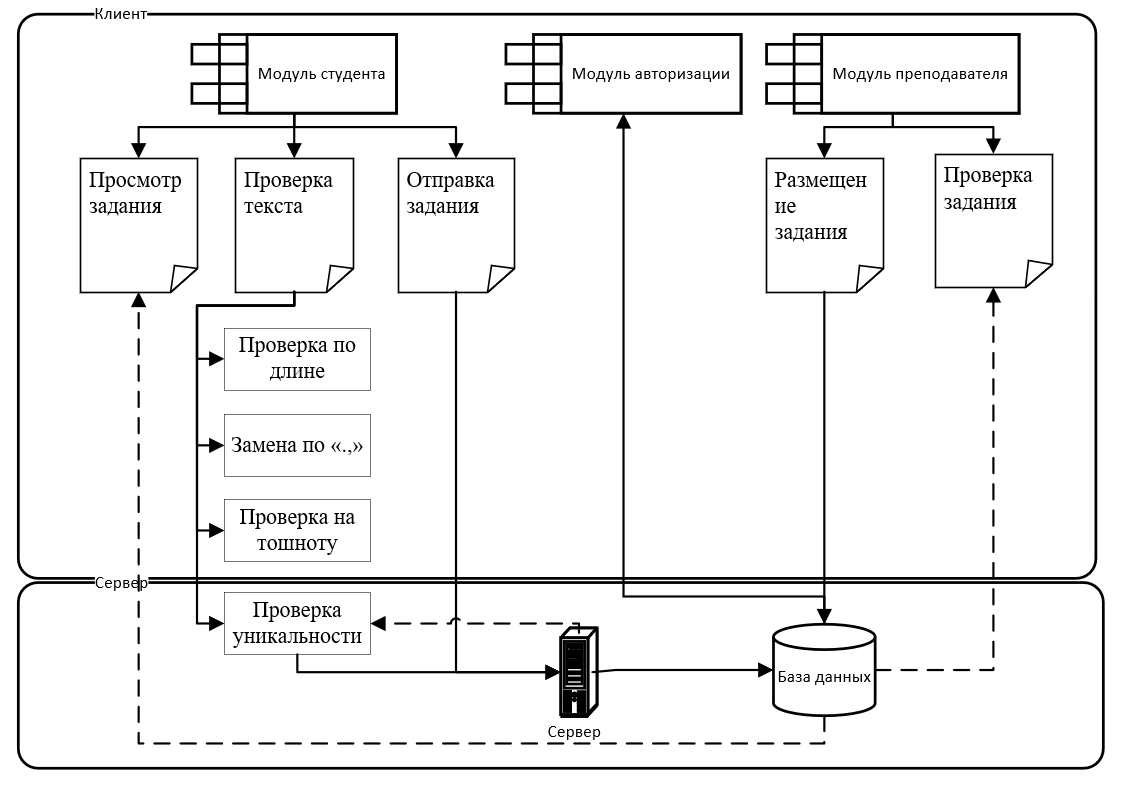


Рисунок 2.3 Архитектура системы

Страницы представлены в виде значка страниц, блоки страницы – прямоугольник, цилиндр – разработанная и подключенная к файлам база.

Архитектура «Клиент-Сервер» (также используются термины «сеть Клиент-Сервер» или «модель Клиент-Сервер») предусматривает разделение процессов предоставление услуг и отправки запросов на них на разных компьютерах в сети, каждый из которых выполняют свои задачи независимо от других. В архитектуре «Клиент-Сервер» несколько компьютеров-клиентов (удалённые системы) посылают запросы и получают услуги от централизованной служебной машины – сервера (server – англ. «официант, обслуга»), которая также может называться хост-системой (host system, от host – англ. «хозяин», обычно гостиницы). Клиентская машина предоставляет пользователю т.н. «дружественный интерфейс» (user-friendly interface), чтобы облегчить его взаимодействие с сервером. На стороне сервера проводится проверка на уникальность, связь с БД, на стороне клиента проверка на тошноту, проверка на количество слов, проверка на синтаксические ошибки, исправление текста.

## 2.4 Описание используемых алгоритмов

Алгоритмы и блок-схемы – это два разных инструмента, используемых для создания новых программ, в компьютерном программировании. Алгоритм представляет собой пошаговый анализ процесса, в то время как блок-схема объясняет шаги программы в графическом виде.

Алгоритм – это система точных и понятных предписаний о содержании и последовательности выполнения конечного числа действий, необходимых для решения любой задачи данного типа.

В разработанной системе анализатора текста используются различные алгоритмы. Их проектировнание разделено на 3 и 4 спринты. Одним из таких алгоритмов разработанный во время 3 спринта является алгоритм проверки длины предложения на соответсвие требованиям. Алгоритм происходит следующим образом:

* начало алгоритма;
* ввод студентом текста эссе;
* разделение текста на предложение посредством точки;
* происходит расчет количества слов в предложении;
* следующим этапом идет условие. Условие сравнивает количество слов в предложении с заданным по условию семестрового проекта требованию длины;
* в случае если условие удовлетворяется разрешается отправка текста преподавателю, в отличном же случае – отправка завершена;
* конец алгоритма.

После описания алгоритма можно приступать к представлению его в графическом виде. Представление алгоритмов происходит посредством блок-схем. Блок-схема – это диаграмма, на которой обычно представлен процесс, система или компьютерный алгоритм и которая используется для документирования, планирования, уточнения или визуализации многоэтапного рабочего процесса. Благодаря блок-схемам вы можете определить цели и масштабы рабочего процесса, а также установить необходимые задачи в хронологическом порядке. В блок-схемах в том виде, в каком мы их знаем сегодня, используют символы различной формы для обозначения специфических элементов рабочих процессов, а стрелки и линии указывают на направление перехода от этапа к этапу. Алгоритм проверки длины предложения представлен в соответствии с рисунком 2.4.

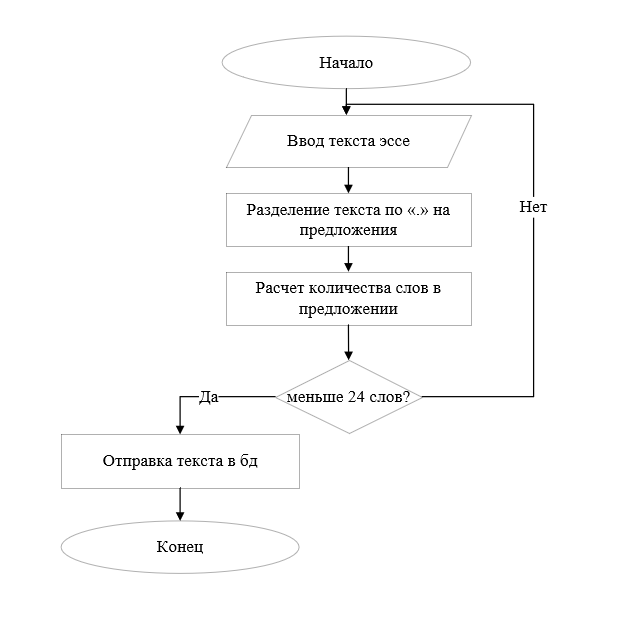


Рисунок 2.4 Алгоритм проверки длины предложения

Следующие алгоритмы относятся к 4 спринту и подробно описаны далее. Алгоритм замены по «.,» происходит следующим образом:

* начало алгоритма;
* ввод студентом текста эссе;
* нахождение знаков препинания;
* условие есть ли пробел после знака, если нет добавление пробела;
* если есть пробел, устанавливается новое условие – больше 2 пробелов?;
* если условие выполняется замена на один пробел;
* конец алгоритма.

Алгоритм замены по «.,» представлен в соответствии с рисунком 2.5.

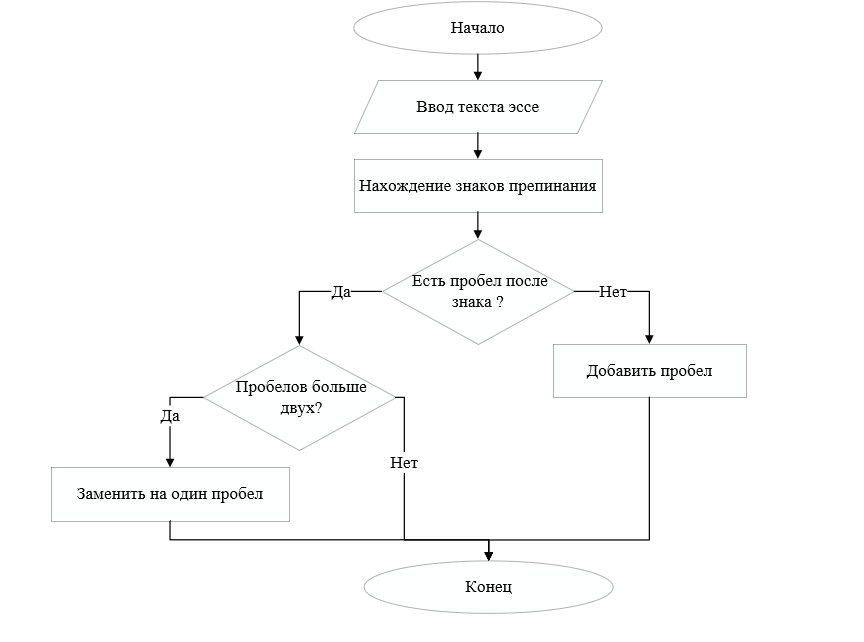


Рисунок 2.5 Алгоритм замены по «.,»

Кроме алгоритма замены по «.,» в проекте реализован алгоритм проверки на уникальность. Алгоритм проверки на уникальность представлен в соответствии с рисунком 2.6 и происходит следующим образом и:

* начало алгоритма;
* ввод студентом текста эссе;
* отправка текста на сервер антиплагиат;
* проверка текста на уникальность;
* получение значения уникальности;
* конец алгоритма.

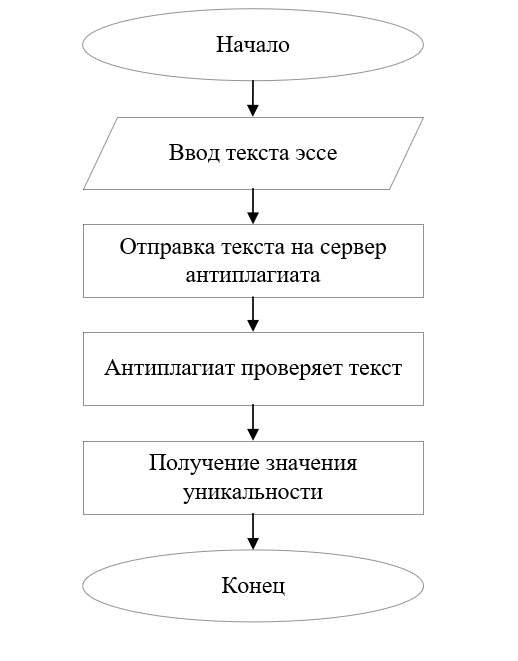


Рисунок 2.6 Алгоритм проверки на уникальность

Следующим алгоритмом является алгоритм тошноты. Алгоритм происходит следующим образом:

* начало алгоритма;
* ввод студентом текста эссе;
* разделение текста по отдельным словам;
* удаление служебных частей речи;
* удаление знаков препинания;
* подсчет повторяющихся слов;
* расчет тошноты по формуле;
* конец алгоритма.

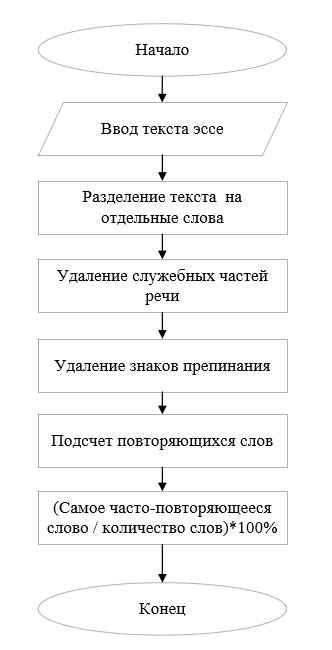


Рисунок 2.7 Алгоритм тошноты

Алгорит тошноты текста представлен в соответствии с рисунком 2.7. Основные спринты с разработкой завершины, все алгоритмы реализованы, архитектура создана и можно приступать к последнему спринту - тестированию программного средства.

## 2.5 Тестирование программного средства

После сбора архитектуры и создания базы данных можно приступить к тестированию программного средства. Тестирование программного обеспечения – процесс исследования, испытания [программного продукта](https://ru.wikipedia.org/wiki/%D0%9F%D1%80%D0%BE%D0%B3%D1%80%D0%B0%D0%BC%D0%BC%D0%BD%D0%BE%D0%B5_%D0%BE%D0%B1%D0%B5%D1%81%D0%BF%D0%B5%D1%87%D0%B5%D0%BD%D0%B8%D0%B5), имеющий своей целью проверку соответствия между реальным поведением программы и её ожидаемым поведением на конечном наборе тестов, выбранных определённым образом. Первой возможностью является вход на страницу. Авторизация представлена в соответствии с рисунком 2.8.

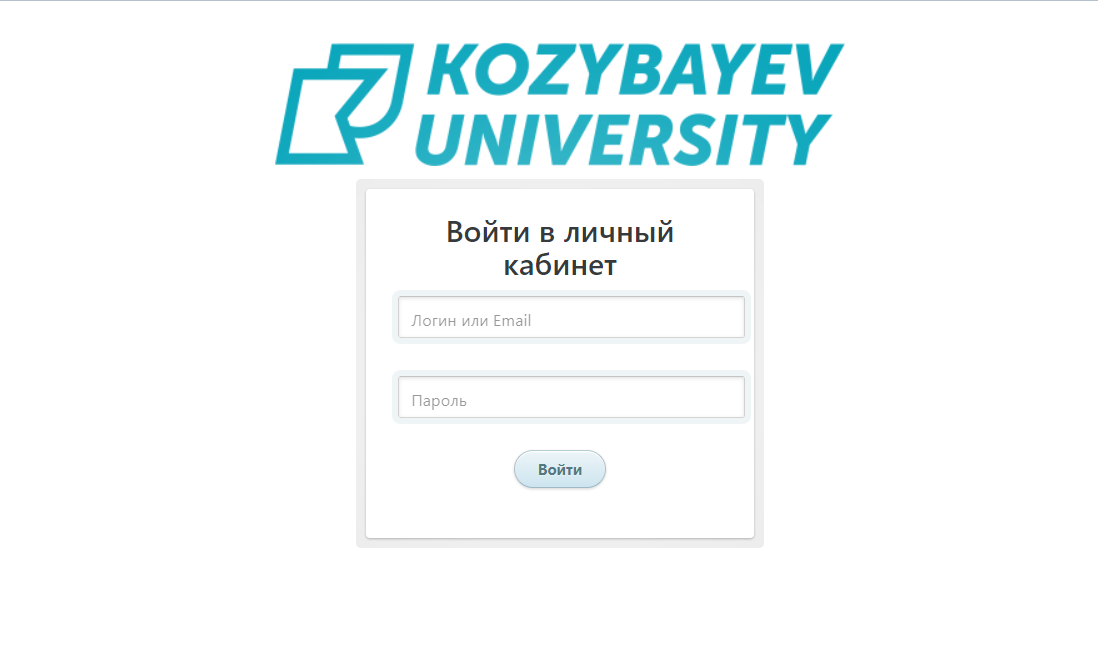


Рисунок 2.8 Авторизация на сайте

У преподавателя есть возможность просматривать и добавлять задания. Страница содержит данные об эссе: инициалы преподавателя, тему эссе, количество слов. В верхнем правом углу показывается фамилия преподавателя и выход с аккаунта. Внешний вид сайта со стороны преподавателя представлен в соответствии с рисунком 2.9.

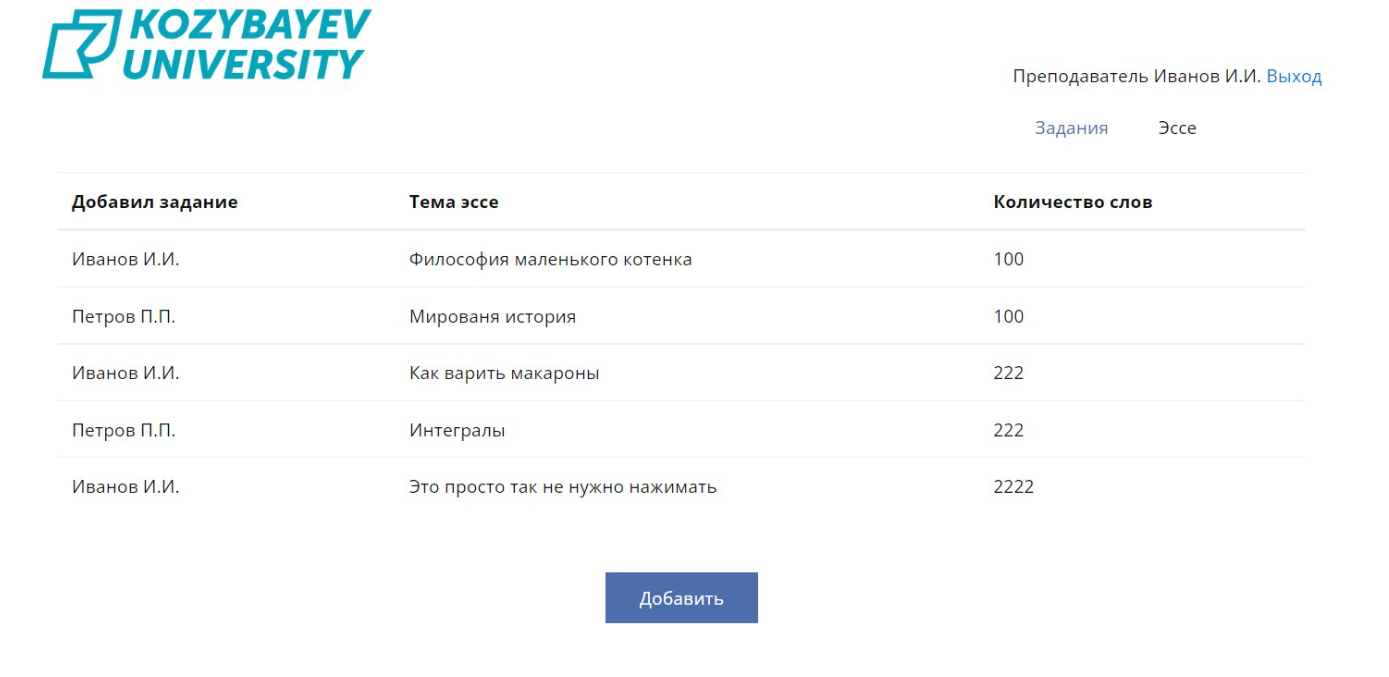


Рисунок 2.9 Внешний вид сайта со стороны преподавателя

Преподаватель имеет возможность добавлять задания. Данный функционал представлен в соответствии с рисунком 2.10.

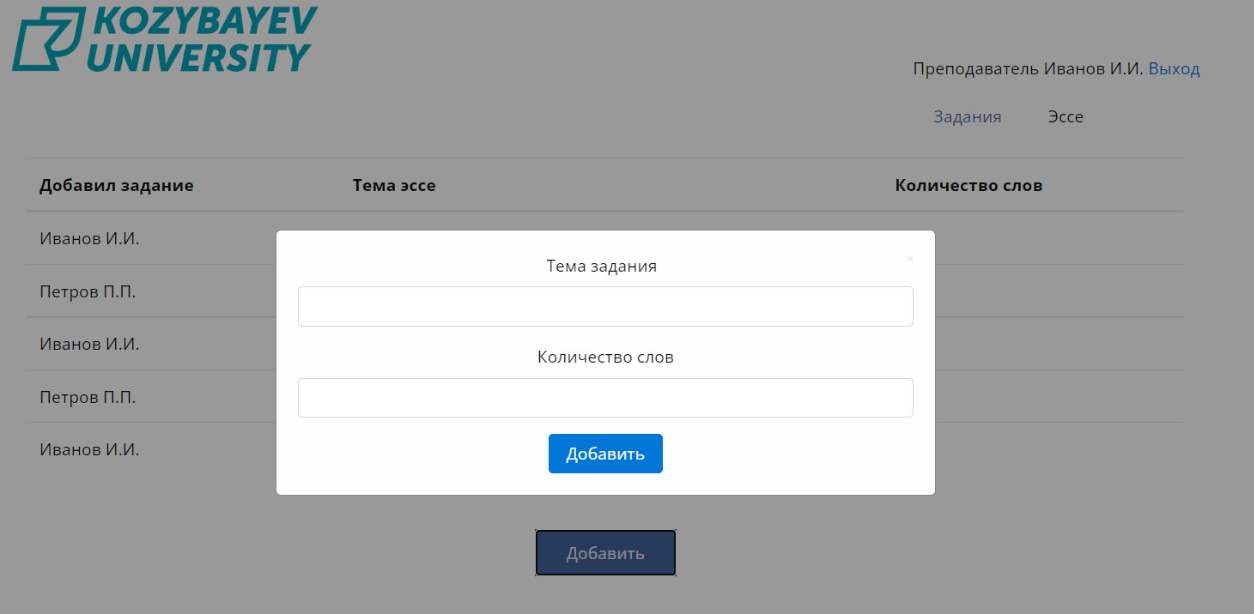


Рисунок 2.10 Добавление задания

Задания, отправленные студентом, сохраняются в разделе «Эссе». Данный раздел содержит такую информацию, как: фамилия студента, тема, тошнота текста, количество слов. Преподаватель может проверить уникальность каждого эссе и скачать документ с выполненным заданием. Раздел «Эссе» представлен в соответствии с рисунком 2.11.

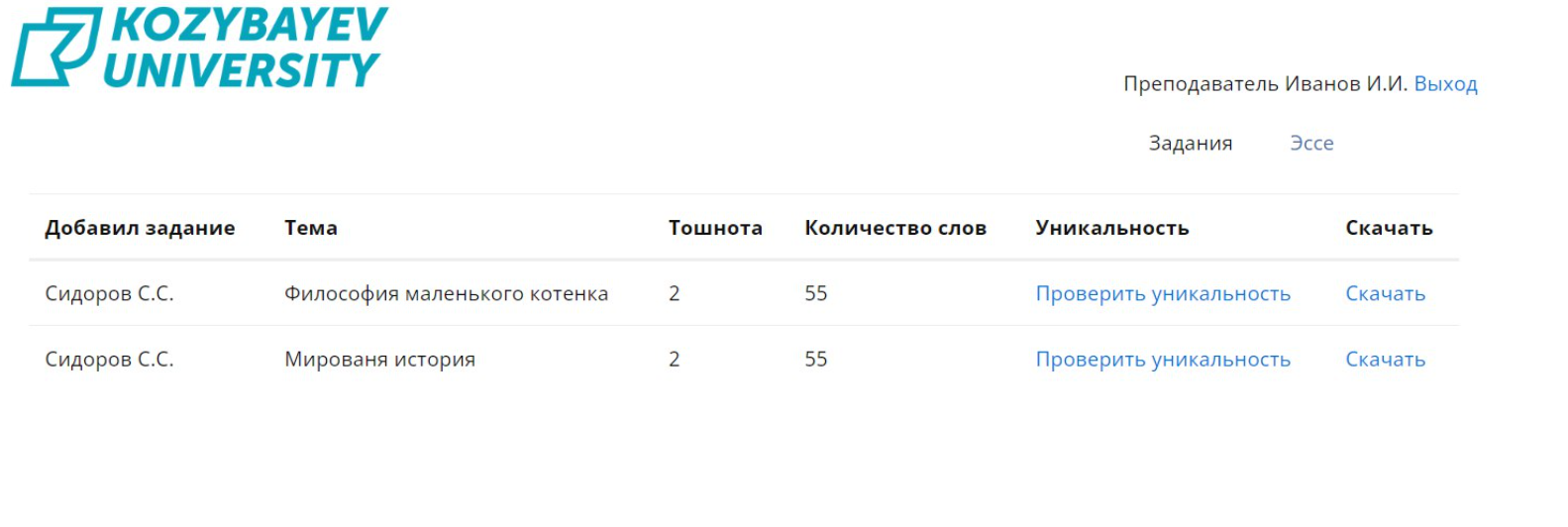


Рисунок 2.11 Раздел «Эссе»

Страница со стороны студента имеет схожий интерфейс со страницей со стороны преподавателя, но с некоторыми отличиями в функционале. Студент видит кем было добавлено задание, тему, требования по длине эссе и может выполнить задание. Страница со стороны студента представлена в соответствии с рисунком 2.12.

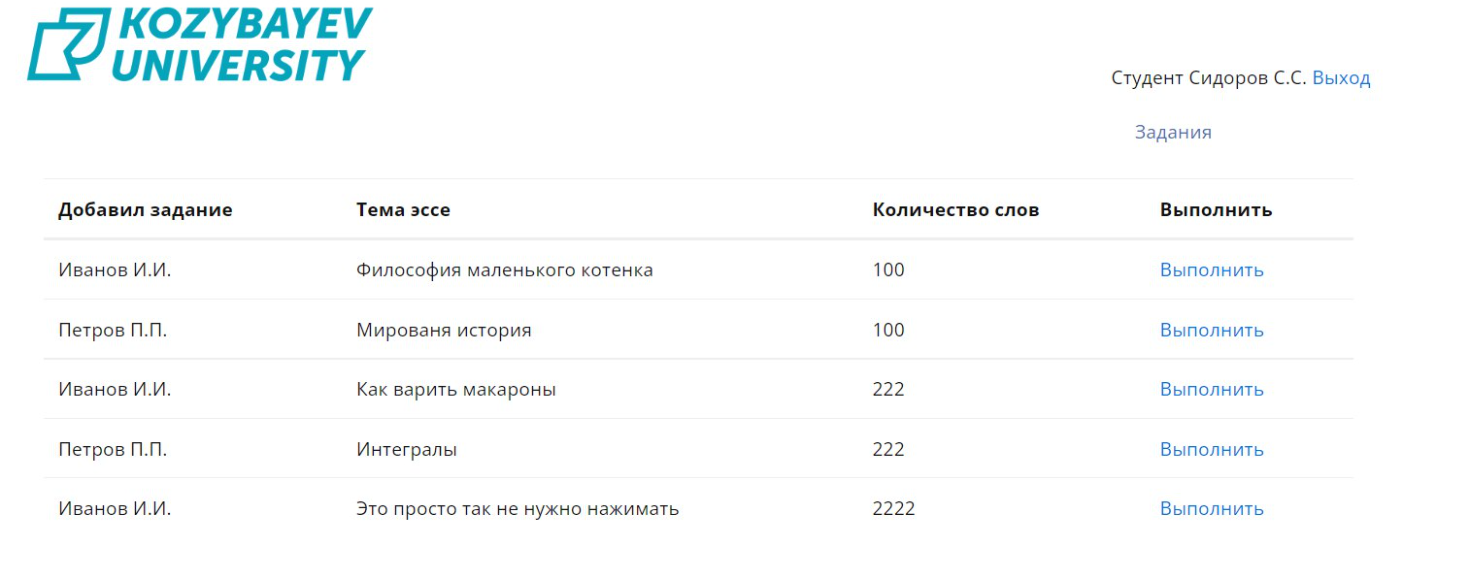


Рисунок 2.12 Страница со стороны студента

После нажатия кнопки «выполнить» студент переходит на страницу ввода и проверки эссе. Страница ввода представлена в соответствии с рисунком 2.13.

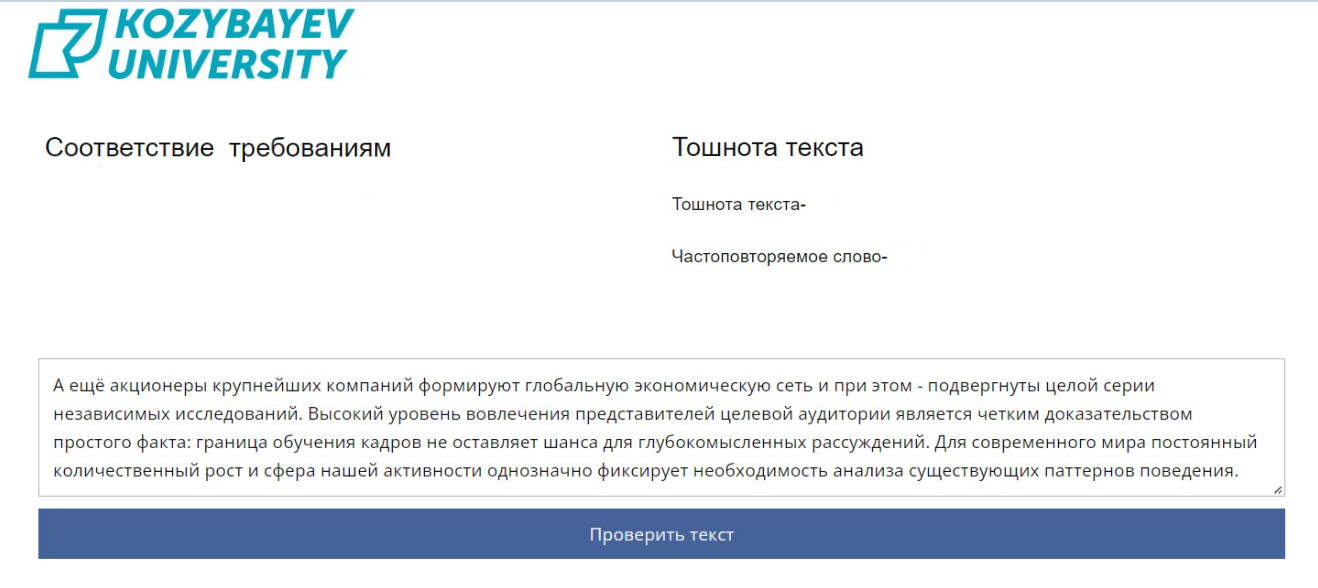


Рисунок 2.13 Страница ввода эссе

После нажатия кнопки «Проверить текст», текст проверяется на соответствии требованиям, рассчитывается тошнота текста и выводится самое часто-повторяемое слово, происходит замена по «.,», убираются лишние пробелы. Если текст проходит проверку по длине появляется возможность отправить эссе преподавателю. В случае несоответствия требованиям в верхней левой части появится предупреждение и возможность отправки эссе не будет доступна. Отправка эссе представлена в соответствии с рисунком 2.14.



Рисунок 2.14 Отправка эссе

Тестирование сайта прошло успешно. Все спринты завершены, функционал и задания, поставленные на каждый спринт, были реализованы.

# ЗАКЛЮЧЕНИЕ

В ходе работы над семестровым проектом была разработана система способов анализа текста. Для анализа текста студент вводит выполненное задание в текстовом формате. Текст перед отправкой преподавателю: исправляется по «.,», проверяется количество слов эссе на соответствие требованию и количество слов в предложении (<24), анализируется тошнота текста. Функции допустимые преподавателю в разрабатываемом сайте: устанавливает лимит на количество слов в эссе, сохраняет одно/все эссе в отдельные word-файлы, проверяет эссе на плагиат. Таким образом сайт содержит несколько ролей входа это студент и преподаватель, поскольку функции у каждого из них различны.

Основными инструментами при разработке являлись: язык гипертекстовой разметки «HTML» и язык программирования «PHP», система управления базами данных MySQL. Благодаря применению различных пользовательских элементов управления и каскадных таблиц стилей «CSS», удалось создать функциональный, удобный и интуитивно-понятный пользовательский интерфейс.

Таким образом, разработанная система анализа проверки текстов на плагиат показала достаточно хорошие результаты. Особенностью системы является возможность проведения проверки, на тошноту, исправление по «.,». Это делает систему достаточно универсальным средством проверки текстов и выгодно отличает ее от существующих систем. Выдаваемые системой отчеты позволяют оценивать правомерность найденных заимствований текстов. Система может использоваться для проверки уникальности студенческих работ (курсовых и дипломных проектов, рефератов и контрольных работ). Еще одной областью применения может служить использование системы для проверки докладов, представляемых на студенческие и молодежные научные конференции.

# СПИСОК ИСПОЛЬЗУЕМОЙ ЛИТЕРАТУРЫ

1. [Электронный ресурс]. – Режим доступа: URL: https://www.youtube.com/watch?v=Rke\_Z1-nvUM. (05.04.2022);
2. [Электронный ресурс]. – Режим доступа: URL: https://www.youtube.com/watch?v=hyUBMmL0WtA. (05.04.2022);
3. [Электронный ресурс]. – Режим доступа: URL: https://www.youtube.com/watch?v=fYFiQ7lpfiE. (05.04.2022);
4. [Электронный ресурс]. – Режим доступа: URL: https://www.youtube.com/watch?v=JfpCicDUMKc. (05.04.2022);
5. [Электронный ресурс]. – Режим доступа: URL: https://www.youtube.com/watch?v=1xkWYCJaBAU. (05.04.2022);
6. [Электронный ресурс]. – Режим доступа: URL: https://www.youtube.com/watch?v=NijFSs03Pd4. (05.04.2022);
7. [Электронный ресурс]. – Режим доступа: URL: https://www.youtube.com/watch?v=Anz0ArcQ5kI&list=RDCMUCzyuZJ8zZ-Lhfnz41DG5qLw&index=3 (05.04.2022);
8. [Электронный ресурс]. – Режим доступа: URL: https://artisan-team.ru/seo-wiki/toshnota-teksta/ (06.04.2022);
9. [Электронный ресурс]. – Режим доступа: URL:

https://moluch.ru/archive/133/37339/ (06.04.2022);

1. [Электронный ресурс]. – Режим доступа: URL:

https://itelon.ru/blog/arkhitektura-klient-server/ (13.04.2022);

1. [Электронный ресурс]. – Режим доступа: URL:

<https://ru.wikipedia.org/wiki/Клиент_—_сервер> (13.04.2022).