**ПРАВИТЕЛЬСТВО РОССИЙСКОЙ ФЕДЕРАЦИИ**

**ФЕДЕРАЛЬНОЕ ГОСУДАРСТВЕННОЕ АВТОНОМНОЕ**

**ОБРАЗОВАТЕЛЬНОЕ УЧРЕЖДЕНИЕ ВЫСШЕГО ОБРАЗОВАНИЯ**

**НАЦИОНАЛЬНЫЙ ИССЛЕДОВАТЕЛЬСКИЙ УНИВЕРСИТЕТ**

**«ВЫСШАЯ ШКОЛА ЭКОНОМИКИ»**

Факультет компьютерных наук

Образовательная программа «Программная инженерия»

|  |  |  |  |
| --- | --- | --- | --- |
|  | СОГЛАСОВАНО  Аспирант факультета компьютерных наук департамента программной инженерии, штатный преподаватель факультета компьютерных наук  \_\_\_\_\_\_\_\_\_\_\_ / Н.А. Терлыч /  «\_\_\_\_» \_\_\_\_\_\_\_\_\_\_\_ 2024 г. | УТВЕРЖДАЮ  Академический руководитель образовательной программы «Программная инженерия»,  старший преподаватель департамента программной инженерии  \_\_\_\_\_\_\_\_\_\_\_ / Н.А. Павлочев /  «\_\_\_\_» \_\_\_\_\_\_\_\_\_\_\_ 2024 г. | |
| |  |  | | --- | --- | | Подп. и дата |  | | Инв. № дубл. |  | | Взам. инв. № |  | | Подп. и дата |  | | Инв. № подл |  | | СИСТЕМА ДЛЯ АВТОМАТИЗАЦИИ ПРОВЕРКИ РУКОПИСНЫХ ОТВЕТОВ  **Техническое задание**  **ЛИСТ УТВЕРЖДЕНИЯ**  **RU.17701729.12.17-01 ТЗ 01-1-ЛУ** | | |
|  | | ИСПОЛНИТЕЛЬ  Студент группы БПИ225  \_\_\_\_\_\_\_\_\_ / А.Е. Дадыков /  «\_\_\_\_» \_\_\_\_\_\_\_\_\_\_\_ 2024 г. |
| Москва 2024 | | |

|  |  |
| --- | --- |
| УТВЕРЖДЕН  RU.17701729.12.17-01 ТЗ 01-1-ЛУ |  |
| |  |  | | --- | --- | | Подп. и дата |  | | Инв. № дубл. |  | | Взам. инв. № |  | | Подп. и дата |  | | Инв. № подл |  | | СИСТЕМА ДЛЯ АВТОМАТИЗАЦИИ ПРОВЕРКИ РУКОПИСНЫХ ОТВЕТОВ  **Техническое задание**  **RU.17701729.12.17-01 ТЗ 01-1**  **Листов 23** | |
| Москва 2024 | |

СОДЕРЖАНИЕ

[АНнотация 4](#_Toc158923289)

[ГЛОССАРИЙ 6](#_Toc158923290)

[1. введение 8](#_Toc158923291)

[1.1 Наименование 8](#_Toc158923292)

[1.2 Краткая характеристика области применения 8](#_Toc158923293)

[2. ОСнования для разработки 9](#_Toc158923294)

[2.1 Документы, на основании которых ведется разработка 9](#_Toc158923295)

[2.2 Наименование темы разработки 9](#_Toc158923296)

[2.3 Условное обозначение темы разработки 9](#_Toc158923297)

[3. назначение разработки 10](#_Toc158923298)

[3.1 Функциональное назначение 10](#_Toc158923299)

[3.2 Эксплуатационное назначение 10](#_Toc158923300)

[4. требования к программе 11](#_Toc158923301)

[4.1 Требования к функциональным характеристикам 11](#_Toc158923302)

[4.1.1 Требования к серверной части 11](#_Toc158923303)

[4.1.2 Требования в клиентской части 12](#_Toc158923306)

[4.2 Требования к надежности 13](#_Toc158923307)

[4.2.1 Требования к обеспечению надежного функционирования программы 13](#_Toc158923308)

[4.2.2 Время восстановления после отказа 13](#_Toc158923309)

[4.3 Условия эксплуатации 14](#_Toc158923310)

[4.3.1 Требования к видам обслуживания 14](#_Toc158923313)

[4.3.2 Требования к численности и квалификации персонала 14](#_Toc158923314)

[4.4 Требования к составу и параметрам технических средств 14](#_Toc158923315)

[4.4.1 Требования к составу и параметрам технических средств клиента 14](#_Toc158923316)

[4.4.2 Требования к составу и параметрам технических средствам сервера 14](#_Toc158923318)

[4.5 Требования к информационной и программной совместимости 15](#_Toc158923320)

[4.5.1 Требования к программным средствам, используемым программой 15](#_Toc158923325)

[4.6 Требования к маркировке и упаковке 15](#_Toc158923326)

[4.7 Требования к транспортированию и хранению 15](#_Toc158923327)

[5. требования к программной документации 16](#_Toc158923328)

[5.1 Состав программной документации 16](#_Toc158923329)

[5.2 Специальные требования к программной документации 16](#_Toc158923330)

[6. технико-экономические показатели 17](#_Toc158923331)

[6.1 Ориентировочная экономическая эффективность 17](#_Toc158923332)

[6.2 Предполагаемая потребность 17](#_Toc158923333)

[6.3 Преимущества разработки по сравнению с отечественными и зарубежными образцами и аналогами 18](#_Toc158923334)

[7. стадии и этапы разработки 19](#_Toc158923335)

[8. порядок контроля и приемки 21](#_Toc158923336)

[8.1 Виды испытаний 21](#_Toc158923337)

[8.2 Общие требования к приемке работы 21](#_Toc158923338)

[СПИСОК ИсточниКОВ 22](#_Toc158923339)

АНнотация

Техническое задание – это основной документ, определяющий цель, набор требований, структуру, свойства, методы и порядок создания программного продукта, в соответствии с которым производится разработка программы, ее тестирование и приемка, исключающий двусмысленность толкования разными исполнителями.

Настоящее техническое задание содержит следующий ряд разделов разделы:

1. «Введение»;
2. «Основания для разработки»;
3. «Назначение разработки»;
4. «Требования к программе»;
5. «Требования к программным документам»;
6. «Технико-экономические показатели»;
7. «Стадии и этапы разработки»;
8. «Порядок контроля и приемки»;
9. «Список источников».

В разделе «Введение» указано наименование и краткая характеристика области применения программы.

В разделе «Основания для разработки» указан документ, на основании которого ведется разработка, и наименование темы разработки.

В разделе «Назначение разработки» указано функциональное и эксплуатационное назначение программного продукта.

Раздел «Требования к программе» содержит основные требования к функциональным характеристикам, к надежности, к условиям эксплуатации, к составу и параметрам технических средств, к информационной и программной совместимости, к маркировке и упаковке, к транспортировке и хранению, а также специальные требования.

Раздел «Требования к программным документам» содержит предварительные состав программной документации и специальные требования к ней.

Раздел «Технико-экономические показатели» содержит ориентировочную экономическую эффективность, предполагаемую годовую потребность, экономические преимущества разработки программы.

Раздел «Стадии и этапы разработки» содержит стадии разработки, этапы и содержание работ.

В разделе «Порядок контроля и приемки» указаны общие требования к приемке работы.

В разделе «Список использованных источников» указаны ссылки на источники, которые были использованы при разработке.

Настоящий документ разработан в соответствии с требованиями:

1. ГОСТ 19.103–77 «Обозначения программ и программных документов» [[1]](#ГОСТ_19_103_77);
2. ГОСТ 19.104–78 «Основные надписи» [[2]](#ГОСТ_19_104_78);
3. ГОСТ 19.105–78 «Общие требования к программным документам» [[3]](#ГОСТ_19_105_78);
4. ГОСТ 19.106-78 «Требования к программным документам, выполненным печатным способом» [[4]](#ГОСТ_19_106_78);
5. ГОСТ 19.201-78 «Техническое задание. Требования к содержанию и оформлению»[[5]](#ГОСТ_19_201_78).

ГЛОССАРИЙ

Таблица 1 – Глоссарий: Используемые термины и определения (алфавитном порядке)

|  |  |
| --- | --- |
| Термин | Определение |
| «Учащийся» | Пользователь программы, который состоит в классе. |
| «Преподаватель» | Пользователь программы, который создает классы и уроки для обучения «Учащихся». |
| Браузер | Программа, которая позволяет просматривать веб-страницы. |
| Класс | Виртуальное пространство, где «Учащиеся» способны просматривать уроки. |
| Клиент | См. термин «Пользователь». |
| Логин | Уникальное имя или идентификатор, который используется пользователем для доступа к программе. |
| Мобильное устройство | Компактные персональные компьютеры с размером диагонали 4-11 дюймов (включая планшетные компьютеры), предназначенные для портативного использования и обладающие возможностью подключения к интернету и камерой. |
| Образовательное учреждение | Учреждение, осуществляющее образовательный процесс, то есть реализующее одну или несколько образовательных программ и (или) обеспечивающее содержание и воспитание обучающихся, воспитанников. |
| Ответ HTTP-запроса (HTTP Response) | Сообщения, которые сервер отправляет в ответ на клиентский HTTP-запрос. |
| Персональный компьютер | Устройства для индивидуального использования, предназначенное для обработки данных, доступа в Интернет, выполнения программ и других задач. |
| Пользователь | Человек, который использует программу по её назначению. |
| Программа | См. термин «Система». |
| Сервер | Компьютер, который обрабатывает запросы от клиентских устройств и предоставляет им доступ к ресурсам и данным. |
| Система | Программное обеспечение, разработанное для автоматизации процесса проверки и оценки рукописных ответов «Учащихся» образовательных учреждений. |
| Урок | Виртуальное пространство, где «Учащиеся» способны смотреть теорию и прикреплять свои рукописные работы в формате фотографий. |
| Фреймворк | Программное обеспечение, которое предоставляет набор инструментов для разработки приложений. |
| API | Набор способов и правил, по которым различные части программы обмениваются между собой данными. |
| FastAPI | Современный фреймворк для создания API на языке Python. |
| Google Classroom | Образовательная платформа для обучения, которая позволяет учителям взаимодействовать с учениками через Интернет. |
| HTTP-запрос (HTTP Request) | Сообщения, которые отправляется клиентом на сервер, чтобы вызвать выполнение некоторых действий. Зачастую для получения доступа к определенному ресурсу. |
| JavaScript | Язык программирования, который в основном используется для разработки клиентской части. |
| JSON | Текстовый формат обмена данными, основанный на JavaScript. |
| Photomath | Мобильное приложение, которое помогает пользователю решать математические задачи, сканируя их с помощью камеры смартфона и предоставляя пошаговые решения. |
| PostgreSQL | Программное обеспечение для сохранения информации о пользователях и классах для корректной работы системы. |
| Pytesseract | Библиотека на языке Python, которая позволяет распознавать текст на изображениях и извлекать его в формате текстовых данных. |
| Python | Язык программирования, использующийся для серверной части. |
| React Native | Современный фреймворк для разработки браузерных и мобильных приложений. |
| Stepik | Образовательная платформа для обучения, которая позволяет учителям взаимодействовать с учениками через Интернет. |

1. введение
   1. Наименование

Полное наименование программы на русском языке: «Система для автоматизации проверки рукописных ответов».

Полное наименование программы на английском языке: «System for Automating the Verification of Handwritten Answers».

Условное наименование программы: «ManuScript».

* 1. Краткая характеристика области применения

Программа обеспечивает автоматическую проверку рукописных работ на соответствие заданным ответам. Программа предоставляет функционал создания учебных классов и уроков, позволяя пользователям загружать и проверять рукописные работы в рамках этих уроков.

Основная цель разрабатываемой программы — ускорить процесс проверки работ и облегчить взаимодействие между преподавателями и учащимися.

1. ОСнования для разработки
   1. Документы, на основании которых ведется разработка

Учебный план подготовки бакалавров по направлению 09.03.04 «Программная инженерия» и утвержденная академическим руководителем тема курсового проекта.

* 1. Наименование темы разработки

Наименование темы разработки на русском языке: «Система для автоматизации проверки рукописных ответов».

Наименование темы разработки на английском языке: «System for Automating the Verification of Handwritten Answers».

* 1. Условное обозначение темы разработки

Условное обозначение темы разработки: «ManuScript».

1. назначение разработки
   1. Функциональное назначение

Программа предоставляет возможность автоматической проверки рукописных работ. Пользователи программы делятся на роли: «Преподаватель» или «Учащийся».

«Преподаватель» имеет возможность создавать классы, а затем в классах создавать уроки. В уроках «Учащиеся» имеют возможность прикрепить свою работу в формате фотографии и автоматически получить оценку работы. Также «Преподаватель» способен прикрепить работу за «Учащегося».

Программа имеет браузерную и мобильную версии.

* 1. Эксплуатационное назначение

Программа ускорит процесс проверки рукописных работ и облегчит взаимодействие между преподавателями и учащимися в образовательных учреждениях.

1. требования к программе
   1. Требования к функциональным характеристикам

Программа состоит из двух основных компонент: серверной и клиентской частей. Эти компоненты взаимодействуют между собой через HTTP-запросы. Серверная часть отвечает за обработку запросов, взаимодействие с базой данных и предоставление API для клиентов. Клиентская часть представлена в виде браузерного и мобильного приложений и обеспечивает пользовательский интерфейс для взаимодействия с сервером.

* + 1. Требования к серверной части

Серверная часть должна предоставлять API для взаимодействия с пользователем и реализовывать следующий функционал:

* После регистрации либо авторизации пользователи программы делятся на роли: «Преподаватель» или «Учащийся». Каждый пользователь может быть «Преподавателем» и «Учащимся» одновременно;
* «Преподаватель» имеет возможность создавать учебные классы, а затем в классах создавать уроки. В уроках «Учащиеся» имеют возможность прикрепить свою работу в формате фотографии и автоматически получить оценку работы;
* «Преподаватель» имеет возможность добавлять в уроки теорию и задания, где «Учащимся» необходимо прикрепить свою работу;
* «Преподаватель» имеет возможность изменить оценку «Учащегося» за урок, если автоматическая проверка неверно определила оценку;
* «Преподаватель» способен добавить работу за «Учащегося»;
* «Преподаватель» имеет возможность посмотреть успеваемость выбранного класса и ученика;
* Каждый пользователь имеет возможность изменить фамилию и имя, почту, пароль;
* Каждый пользователь имеет возможность выбрать один из доступных языков программы.
  + 1. Требования в клиентской части

Клиентская часть должна быть реализована в виде браузерного и мобильного приложений и предоставлять интерфейс для взаимодействия с серверной частью. Должны быть реализованы:

1. **Страница регистрации / авторизации**: на этой странице пользователи могут создать новый аккаунт в системе (регистрация), введя логин, пароль и фамилию с именем, или войти в свой существующий аккаунт (авторизация), введя логин и пароль.
2. **Страница профиля**: эта страница отображает информацию о пользователе: фамилию, имя, электронную почту, изображение профиля. Пользователь может просматривать и редактировать свой профиль на этой странице.
3. **Страница настроек**: на этой пользователь может настраивать параметры своего аккаунта и предпочтения, такие как уведомления и безопасность (смена логина, пароля, фамилии и имени).
4. **Страница учебных классов для учащегося**: на этой странице отображаются все учебные классы, в которых зарегистрирован текущий пользователь в роли учащегося. Он может видеть список доступных классов, присоединяться к новым и управлять уже существующими (закрепить, удалить).
5. **Конструктор учебных классов**: интерфейс, который позволяет преподавателю создавать новые учебные классы. Пользователь может добавлять уроки, учеников, учителей, устанавливать параметры класса и т. д.
6. **Страница учебного класса**: на этой странице пользователь может просматривать информацию о конкретном учебном классе, такую как список участников, материалы, задания и прочее.
7. **Страница урока**: эта страница позволяет просматривать содержание конкретного урока в рамках выбранного учебного класса. Здесь могут быть представлены материалы, задания, домашние задания.
8. **Страница уведомлений**: на этой странице пользователь может просматривать уведомления, связанные с его аккаунтом, учебными классами или уроками, такие как новые сообщения, обновления.

Мобильная и браузерная версии должны иметь одинаковый функционал программы, за исключением измененного размера экрана.

* 1. Требования к надежности
     1. Требования к обеспечению надежного функционирования программы

Функционал, предоставленный пользователю, не должен приводить к аварийному завершению программы или возникновению серверных ошибок.

* + 1. Время восстановления после отказа

Время восстановления после отказа, вызванного сбоем технических средств, соответствует времени, необходимому на перезапуск программы.

* 1. Условия эксплуатации
     1. Требования к видам обслуживания

Серверный компьютер, обеспечивающий работу серверной части программы, должен периодические проходить проверку на предмет выявления сбоев, неполадок и отсутствия вредоносных программ.

* + 1. Требования к численности и квалификации персонала

Требования к численности и квалификации персонала регулируются в соответствии с Руководством Оператора.

* 1. Требования к составу и параметрам технических средств
     1. Требования к составу и параметрам технических средств клиента

Минимальные требования к составу и параметрам технических средствам клиента для полноценного использования программы:

1. Для полноценной эксплуатации программы на персональном компьютере: пользователь должен обеспечить соответствие минимальным требованиям, установленным для последней версии браузера.
2. Для полноценной эксплуатации программы на мобильном устройстве: пользователь должен обеспечить соответствие минимальным требованиям, установленным для последней версии операционной системы.
   * 1. Требования к составу и параметрам технических средствам сервера

Минимальные требования к составу и параметрам технических средствам сервера являются:

* Процессор: Intel Core i5-7xxx или эквивалентный процессор с поддержкой параллельный вычислений;
* Оперативная память (RAM): 8 ГБ оперативной памяти;
* Хранилище данных: жесткий диск или SSD-накопитель 128 ГБ;
* Сетевой адаптер: 100Мбит/с.
  1. Требования к информационной и программной совместимости
     1. Требования к программным средствам, используемым программой

Требования для клиента: браузер или установленное мобильное приложение.

Требования для сервера: стабильная версия Python, установленный фреймворк FastAPI.

* 1. Требования к маркировке и упаковке

Требования к маркировке и упаковке не предъявляются.

* 1. Требования к транспортированию и хранению

Специальные требования к транспортировке и хранению не предъявляются.

1. требования к программной документации
   1. Состав программной документации
2. «ManuScript». Техническое задание (по ГОСТ 19.201-78);
3. «ManuScript». Пояснительная записка (по ГОСТ 19.404-79);
4. «ManuScript». Программа и методика испытаний (по ГОСТ 19.301-79);
5. «ManuScript». Руководство оператора (по ГОСТ 19.505-79);
6. «ManuScript». Текст программы (по ГОСТ 19.401-78).
   1. Специальные требования к программной документации

Документы к программе должны быть выполнены в соответствии с ГОСТ 19.106-78 и ГОСТами по каждому виду документа (см. п. 5.1).

Пояснительная записка должны быть загружена в систему Антиплагиата через LMS «НИУ ВШЭ».

Техническое задание и пояснительная записка, титульные листы других документов должны быть подписаны руководителем разработки и исполнителем.

Документация и программа сдается в электронном виде в формате .pdf или .docx в архиве формата .zip или .rar.

За три дня до защиты проекта комиссии все материалы курсового проекта:

* Программная документация;
* Программный проект;
* Исполняемый файл;
* Отзыв руководителя;
* Отчет системы Антиплагиат.

должны быть загружены одним или несколькими архивами в проект дисциплины «Курсовой проект» в личном кабинете в информационной образовательной среде SmartLMS НИУ ВШЭ.

1. технико-экономические показатели
   1. Ориентировочная экономическая эффективность

В рамках данной работы расчёт экономической эффективности не предусмотрен.

* 1. Предполагаемая потребность

На данный момент не существует прямых аналогов разрабатываемой системы. Наиболее широко для взаимодействия преподавателей и учащихся используется образовательная платформа Stepik. Данная платформа предоставляет возможности создания учебных классов, уроков и оценивания учащихся через стандартизированные тесты. Разрабатываемая система позволит автоматически оценивать учащихся через фотографии их рукописных работ.

* 1. Преимущества разработки по сравнению с отечественными и зарубежными образцами и аналогами

Таблица 2 – Сравнительная таблица аналогов

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Функциональность | Google Classroom | Photomath | Stepik | ManuScript |
| Сканирование текста с фотографии | – | + | – | + |
| Автоматическая проверка рукописных работ учащихся | – | – | – | + |
| Автоматическая проверка через стандартизированные тесты | + | – | + | – |
| Возможность создавать классы и уроки | + | – | + | + |

Из сравнительной таблицы аналогов ManuScript выделяется преимуществом в виде автоматической проверки рукописных работ учащихся. Эта функция отсутствует у других аналогов, что делает ManuScript уникальным решением в области образовательных технологий.

1. стадии и этапы разработки

Таблица 3 – Стадии и этапы разработки

|  |  |  |  |
| --- | --- | --- | --- |
| Стадия разработки | Этап разработки | Содержание работы | Срок |
| Техническое задание | Подготовительные работы | Постановка задачи | 15 февраля  2024 |
| Анализ аналогов |
| Выбор технологий |
| Сбор теоретических материалов |
| Разработка и утверждение технического задания | Определение требований |
| Определение стадий и этапов разработки |
| Согласование и утверждение технического задания |
| Рабочий проект | Разработка программы | Программирование и отладка программы | февраль -апрель  2024 |
| Разработка программной документации | Разработка программных документов в соответствии с ГОСТ 19.101.77 |
| Тестирование программы | Проведение функционального тестирования |
| Проведение нагрузочного тестирования |
| Корректировка программы и программной документации по результатам испытаний |
| Защита проекта | Загрузка итоговых документов | Подготовка программ и программной документации | 2 апреля  2024 |
| Получение отзыва от научного руководителя |
| Загрузка полного комплекта документации по проекту в SmartLMS |
| Защита проекта | Защита программного проекта (курсового проекта) комиссии | 17-24  апреля  2024 |

1. порядок контроля и приемки
   1. Виды испытаний

Производится проверка корректного выполнения программой заложенных в нее функций, т.е. осуществляется функциональное тестирование программы. Также осуществляется визуальная проверка интерфейса программы на соответствие пункту 4.1. настоящего технического задания. Функциональное тестирование осуществляется в соответствии с документом “«ManuScript». Программа и методика испытаний (ГОСТ 19.301-79)”, в котором указывают:

1. перечень функций программы, выделенных в программе для испытаний, и перечень требований, которым должны соответствовать эти функции;
2. перечень необходимой документации и требования к ней;
3. методы испытаний и обработки информации;
4. технические средства и порядок проведения испытаний;
   1. Общие требования к приемке работы

Прием программы будет утвержден при корректной работе программы в соответствии с пунктом 4.1 при различных входных данных и при предоставлении полной документации к продукту, выполненной в соответствии с требованиями, указанными в пунктах 5.1 и 5.2 данного технического задания.

СПИСОК ИсточниКОВ

1. ГОСТ 19.103–77 «Обозначения программ и программных документов» // Единая система про­граммной документации. – М.: ИПК Издательство стандартов, 2001;
2. ГОСТ 19.104–78 «Основные надписи» // Единая система про­граммной документации. – М.: ИПК Издательство стандартов, 2001;
3. ГОСТ 19.105–78 «Общие требования к программным документам» // Единая система про­граммной документации. – М.: ИПК Издательство стандартов, 2001;
4. ГОСТ 19.106-78 «Требования к программным документам, выполненным печатным способом» // Единая система про­граммной документации. – М.: ИПК Издательство стандартов, 2001.;
5. ГОСТ 19.201-78 «Техническое задание. Требования к содержанию и оформлению» // Единая система про­граммной документации. – М.: ИПК Издательство стандартов, 2001.;
6. ГОСТ 19.301-79 «Программа и методика испытаний. Требования к содержанию и оформлению» // Единая система про­граммной документации. – М.: ИПК Издательство стандартов, 2001.;
7. ГОСТ 19.401-78 «Текст программы. Требования к содержанию и оформлению» // Единая система про­граммной документации. – М.: ИПК Издательство стандартов, 2001.;
8. ГОСТ 19.404-79 «Пояснительная записка. Требования к содержанию и оформлению» // Единая система про­граммной документации. – М.: ИПК Издательство стандартов, 2001.;
9. ГОСТ 19.505-79 «Руководство оператора. Требования к содержанию и оформлению» // Единая система про­граммной документации. – М.: ИПК Издательство стандартов, 2001.;
10. Python: <https://www.python.org/> (дата обращения: 07.02.2024);
11. Pytesseract: <https://pypi.org/project/pytesseract/> (дата обращения: 07.02.2024);
12. FastAPI: <https://fastapi.tiangolo.com/> (дата обращения: 07.02.2024);
13. JavaScript: <https://ru.wikipedia.org/wiki/JavaScript> (дата обращения: 07.02.2024);
14. React Native: <https://reactnative.dev/> (дата обращения: 07.02.2024);
15. PostgreSQL: <https://www.postgresql.org/> (дата обращения: 07.02.2024);
16. Stepik: <https://stepik.org/> (дата обращения: 07.02.2024);
17. Google Classroom: <https://edu.google.com/workspace-for-education/classroom/> (дата обращения: 07.02.2024);
18. Photomath: <https://photomath.com/> (дата обращения: 07.02.2024);