

Школа – Инженерная школа информационных технологий и робототехники  
 Направление подготовки – 09.03.04 Программная инженерия  
 ООП – Разработка программно-информационных систем  
 Отделение школы (НОЦ) – Отделение информационных технологий

### ВЫПУСКНАЯ КВАЛИФИКАЦИОННАЯ РАБОТА БАКАЛАВРА

Тема работы
<b>Разработка учебной онлайн-платформы с визуализацией известных алгоритмов</b>
УДК 004.415.2:378.147:004.421

Обучающийся

Группа	ФИО	Подпись	Дата
8К93	Мамытов Даир		

Руководитель ВКР

Должность	ФИО	Ученая степень, звание	Подпись	Дата
Доцент ОИТ	Мыцко Евгений Алексеевич	к.т.н.		

### КОНСУЛЬТАНТЫ ПО РАЗДЕЛАМ:

По разделу «Финансовый менеджмент, ресурсоэффективность и ресурсосбережение»

Должность	ФИО	Ученая степень, звание	Подпись	Дата
Профессор ОСГН	Гасанов Магеррам Али оглы	д.э.н.		

По разделу «Социальная ответственность»

Должность	ФИО	Ученая степень, звание	Подпись	Дата
Старший преподаватель ООД	Мезенцева Ирина Леонидовна			

### ДОПУСТИТЬ К ЗАЩИТЕ:

Руководитель ООП/ОПОП, должность	ФИО	Ученая степень, звание	Подпись	Дата
Доцент ОИТ	Чердынцев Евгений Сергеевич	к.т.н.		

## ПЛАНИРУЕМЫЕ РЕЗУЛЬТАТЫ ОСВОЕНИЯ ООП/ОПОП

Код компетенции	Наименование компетенции
<b>Универсальные компетенции</b>	
<b>УК(У)-1</b>	Способен осуществлять поиск, критический анализ и синтез информации, применять системный подход для решения поставленных задач.
<b>УК(У)-2</b>	Способен определять круг задач в рамках поставленной цели и выбирать оптимальные способы их решения, исходя из действующих правовых норм, имеющихся ресурсов и ограничений.
<b>УК(У)-3</b>	Способен осуществлять социальное взаимодействие и реализовывать свою роль в команде.
<b>УК(У)-4</b>	Способен осуществлять деловую коммуникацию в устной и письменной формах на государственном языке Российской Федерации и иностранном(-ых) языке(-ах).
<b>УК(У)-5</b>	Способен воспринимать межкультурное разнообразие общества в социально-историческом, этическом и философском контекстах
<b>УК(У)-6</b>	Способен управлять своим временем, выстраивать и реализовывать траекторию саморазвития на основе принципов образования в течение всей жизни.
<b>УК(У)-7</b>	Способен поддерживать должный уровень физической подготовленности для обеспечения полноценной социальной и профессиональной деятельности.
<b>УК(У)-8</b>	Способен создавать и поддерживать безопасные условия жизнедеятельности, в том числе при возникновении чрезвычайных ситуаций.
<b>УК(У)-9</b>	Способен проявлять предприимчивость в практической деятельности, в т.ч. в рамках разработки коммерчески перспективного продукта на основе научно-технической идеи.
<b>Общепрофессиональные компетенции</b>	
<b>ОПК(У)-1</b>	Способен применять естественнонаучные и общепрофессиональные знания, методы математического анализа и моделирования, теоретического и экспериментального исследования в профессиональной деятельности.
<b>ОПК(У)-2</b>	Способен использовать современные информационные технологии и программные средства, в том числе отечественного производства, при решении задач профессиональной деятельности.
<b>ОПК(У)-3</b>	Способен решать стандартные задачи профессиональной деятельности на основе информационной и библиографической культуры с применением информационно-коммуникационных технологий и с учетом основных требований информационной безопасности.
<b>ОПК(У)-4</b>	Способен участвовать в разработке стандартов, норм и правил, а также технической документации, связанной с профессиональной деятельностью.
<b>ОПК(У)-5</b>	Способен устанавливать программное и аппаратное обеспечение для информационных и автоматизированных систем.

<b>ОПК(У)-6</b>	Способен разрабатывать алгоритмы и программы, пригодные для практического использования, применять основы информатики и программирования к проектированию, конструированию и тестированию программных продуктов.
<b>ОПК(У)-7</b>	Способен применять в практической деятельности основные концепции, принципы, теории и факты, связанные с информатикой.
<b>ОПК(У)-8</b>	Способен осуществлять поиск, хранение, обработку и анализ информации из различных источников и баз данных, представлять ее в требуемом формате с использованием информационных, компьютерных и сетевых технологий.
<b>Профессиональные компетенции</b>	
<b>ПК(У)-1</b>	Способен выполнять интеграцию программных модулей и компонент.
<b>ПК(У)-2</b>	Владение навыками моделирования, анализа и использования формальных методов конструирования программного обеспечения.
<b>ПК(У)-3</b>	Способен создавать техническую документацию на продукцию в сфере информационных технологий, управлять технической информацией.
<b>ПК(У)-4</b>	Владение навыками использования операционных систем, сетевых технологий, средств разработки программного интерфейса, применения языков и методов формальных спецификаций, систем управления базами данных.
<b>ПК(У)-5</b>	Способен проводить, оценивать и следить за выполнением концептуального, функционального и логического проектирования систем малого и среднего масштаба и сложности.

Школа – Инженерная школа информационных технологий и робототехники  
 Направление подготовки – 09.03.04 Программная инженерия  
 Отделение школы (НОЦ) – Отделение информационных технологий

УТВЕРЖДАЮ:  
 Руководитель ООП  
 \_\_\_\_\_ Чердынцев Е.С.  
 (Подпись) (Дата) (Ф.И.О.)

**ЗАДАНИЕ  
на выполнение выпускной квалификационной работы**

Обучающийся:

Группа	ФИО
8K93	Мамытов Даир

Тема работы:

<b>Разработка учебной онлайн-платформы с визуализацией известных алгоритмов</b>	
Утверждена приказом директора (дата, номер)	№ 30-96/с от 30.01.2023 г.

Срок сдачи студентом выполненной работы:	29.05.2023
--	------------

**ТЕХНИЧЕСКОЕ ЗАДАНИЕ:**

<b>Исходные данные к работе</b>	Техническое задание к реализации web-приложения образовательного характера для дистанционного изучения дисциплины «Алгоритмы и анализ сложности»
<b>Перечень подлежащих исследованию, проектированию и разработке вопросов</b>	<ul style="list-style-type: none"> <li>– анализ предметной области;</li> <li>– выбор средств разработки для онлайн-платформы;</li> <li>– разработка механизма визуализации алгоритмов;</li> <li>– интеграция компилятора для работы студентов по освоению алгоритмов;</li> <li>– реализация функции выгрузки отчетов по проделанной работе.</li> </ul>
<b>Перечень графического материала</b>	<ul style="list-style-type: none"> <li>– диаграмма вариантов использования;</li> <li>– диаграмма последовательности;</li> <li>– модель базы данных;</li> <li>– диаграмма пакетов web-приложения в нотации</li> </ul>

<b>Консультанты по разделам выпускной квалификационной работы</b>	
<b>Раздел</b>	<b>Консультант</b>
Социальная ответственность	Мезенцева Ирина Леонидовна, старший преподаватель ООД
Финансовый менеджмент, ресурсоэффективность и ресурсосбережение	Гасанов Магеррам Али оглы, профессор ОСГН, д.э.н.

<b>Дата выдачи задания на выполнение выпускной квалификационной работы по линейному графику</b>	23.01.2023
---	------------

**Задание выдал руководитель:**

<b>Должность</b>	<b>ФИО</b>	<b>Ученая степень, звание</b>	<b>Подпись</b>	<b>Дата</b>
Доцент ОИТ	Мыцко Евгений Алексеевич	К.Т.Н.		

**Задание принял к исполнению обучающийся:**

<b>Группа</b>	<b>ФИО</b>	<b>Подпись</b>	<b>Дата</b>
8К93	Мамытов Даир		

Школа – Инженерная школа информационных технологий и робототехники  
 Направление подготовки – 09.03.04 «Программная инженерия»  
 Уровень образования – Бакалавриат  
 Отделение школы (НОЦ) – Отделение информационных технологий  
 Период выполнения – весенний семестр 2022 /2023 учебного года

**КАЛЕНДАРНЫЙ РЕЙТИНГ-ПЛАН  
выполнения выпускной квалификационной работы**

Обучающийся:

<b>Группа</b>	<b>ФИО</b>
8К93	Мамытов Даир

Тема работы:

Разработка учебной онлайн-платформы с визуализацией известных алгоритмов
--

Срок сдачи студентом выполненной работы:	29.05.2023
--	------------

Дата контроля	Название раздела (модуля) / вид работы (исследования)	Максимальный балл раздела (модуля)
27.02.23	Анализ предметной области	20
10.03.23	Проектирование системы	20
20.05.23	Разработка онлайн-платформы	40
13.05.23	Финансовый менеджмент, ресурсоэффективность и ресурсосбережение	10
13.05.23	Социальная ответственность	10

**СОСТАВИЛ:**

**Руководитель ВКР**

Должность	ФИО	Ученая степень, звание	Подпись	Дата
Доцент ОИТ	Мыцко Евгений Алексеевич	К.Т.Н.		

**СОГЛАСОВАНО:**

**Руководитель ООП**

Должность	ФИО	Ученая степень, звание	Подпись	Дата
Доцент ОИТ	Чердынцев Евгений Сергеевич	К.Т.Н.		

**Обучающийся**

Группа	ФИО	Подпись	Дата
8К93	Мамытов Даир		

## Реферат

Пояснительная записка содержит 87 страниц (без учета приложений), 31 рисунок, 22 таблиц, 26 источников, 1 приложение.

Ключевые слова: веб-приложение, онлайн-платформа, программирование, база данных, веб-сайт, фреймворк, алгоритм, компилятор.

Работа посвящена разработке онлайн-платформы для дистанционного изучения дисциплины «Алгоритмы и анализ сложности». Система позволяет студентам изучать работу известных алгоритмов сортировки, поиска элементов, обхода дерева, вычисления контрольной суммы на основе визуализированных примеров. Также система позволяет выполнять простые упражнения в онлайн-среде с формированием отчета о проделанной работе.

Объект исследования – онлайн-платформа для дистанционного обучения.

Цель работы – разработка онлайн-платформы для дистанционного изучения дисциплины «Алгоритмы и анализ сложности».

В результате работы реализовано веб-приложение, позволяющее изучить работу представленных алгоритмов, написать и проверить написанный код в интегрированном в веб-сайт компиляторе с формированием отчета о проделанной работе. Значимость результатов работы заключается в предоставлении пользователям без участия преподавателя изучить материал наглядно на визуализированном примере, а сгенерированный отчет способен сэкономить время на выполнение работы. Также разработанное решение частично снимает нагрузку с преподавателя дисциплины.

## СОДЕРЖАНИЕ

ВВЕДЕНИЕ .....	11
ОБЗОР ЛИТЕРАТУРЫ .....	14
1. ЗАДАЧА ОНЛАЙН-ПЛАТФОРМЫ .....	15
1.1. Обзор аналогов .....	15
2. ПРОЕКТИРОВАНИЕ СИСТЕМЫ .....	17
2.1. Варианты использования .....	17
2.2. Диаграммы последовательности для системы .....	18
2.3. Архитектура системы .....	19
2.4. Выбор технологий для реализации .....	20
3. ПРОГРАММНАЯ РЕАЛИЗАЦИЯ ОНЛАЙН-ПЛАТФОРМЫ .....	26
3.1. Разработка базы данных информационной системы .....	26
3.2. Разработка серверной части приложения .....	28
3.3. Тестирование серверной части приложения .....	31
3.4. Разработка клиентской части приложения .....	33
ЗАДАНИЕ ДЛЯ РАЗДЕЛА «ФИНАНСОВЫЙ МЕНЕДЖМЕНТ, РЕСУРСОЭФФЕКТИВНОСТЬ И РЕСУРСОСБЕРЕЖЕНИЕ» .....	45
4. ФИНАНСОВЫЙ МЕНЕДЖМЕНТ, РЕСУРСОЭФФЕКТИВНОСТЬ И РЕСУРСОСБЕРЕЖЕНИЕ .....	46
4.1. Оценка коммерческого потенциала и перспективности проведения научных исследований с позиции ресурсоэффективности и ресурсосбережений .....	46
4.1.1. Потенциальные потребители результатов исследования .....	46
4.1.2. Анализ конкурентных технических решений .....	47
4.1.3. Технология QuaD .....	48
4.1.4. SWOT-анализ .....	49
4.2. Инициация проекта .....	51
4.2.1. Цели и результаты проекта .....	51
4.3. Планирование научно-исследовательских работ .....	53
4.3.1. Структура работ в рамках научного исследования .....	53



4.3.2.	Определение трудоемкости выполнения работ.....	54
4.3.3.	Определение трудоемкости выполнения работ.....	55
4.3.4.	Бюджет научно-технического исследования (НТИ) .....	58
4.3.4.1.	Расчет материальных затрат НТИ.....	58
4.3.4.2.	Расчет затрат на специальное оборудование для научных (экспериментальных) затрат .....	59
4.3.4.3.	Основная заработная плата исполнителей темы .....	60
4.3.4.4.	Дополнительная заработная плата исполнителей темы .....	62
4.3.4.5.	Отчисления во внебюджетные фонды (страховые отчисления) 63	
4.3.4.6.	Накладные расходы .....	63
4.3.4.7.	Формирование бюджета затрат научно-технического проекта	64
4.4.	Определение ресурсной (ресурсосберегающей), финансовой, бюджетной, социальной и экономической эффективности исследования .	65
	ЗАДАНИЕ ДЛЯ РАЗДЕЛА «СОЦИАЛЬНАЯ ОТВЕТСТВЕННОСТЬ».....	68
5.	СОЦИАЛЬНАЯ ОТВЕТСТВЕННОСТЬ.....	70
5.1.	Введение .....	70
5.2.	Правовые и организационные вопросы обеспечения безопасности .	71
5.2.1.	Правовые нормы .....	71
5.2.2.	Эргономические требования к правильному расположению и компоновке рабочей зоны .....	72
5.3.	Производственная безопасность .....	73
5.3.1.	Отсутствие или недостаток необходимой освещенности .....	74
5.3.2.	Умственное перенапряжение, в том числе вызванное информационной нагрузкой.....	75
5.3.3.	Превышение уровня шума и другие его неблагоприятные характеристики.....	76
5.3.4.	Отклонение показателей микроклимата.....	77
5.3.5.	Производственные факторы, связанные с электрическим током	78
5.4.	Экологическая безопасность .....	79

5.5. Безопасность в чрезвычайных ситуациях .....	80
5.6. Вывод .....	82
ЗАКЛЮЧЕНИЕ .....	84
СПИСОК ИСПОЛЬЗОВАННЫХ ИСТОЧНИКОВ .....	85
ПРИЛОЖЕНИЕ А .....	88

## ВВЕДЕНИЕ

Алгоритмы являются основой языков программирования. Это набор инструкций, которые сообщают компьютеру, как выполнять определенную задачу. В сегодняшнюю цифровую эпоху алгоритмы используются повсеместно, от простых расчетов до сложного анализа данных. Таким образом, изучение алгоритмов имеет решающее значение для любого программиста, который хочет создавать эффективные и действенные приложения.

Одна из основных причин важности изучения алгоритмов заключается в том, что помогает разработчикам понять, как решать проблемы. Будь то небольшая задача или сложная, понимание алгоритма помогает программисту разбить ее на более мелкие, более управляемые части. Это упрощает программирование и отладку.

Кроме того, изучение алгоритмов позволяет программистам оптимизировать свой код для повышения производительности. Производительность является критическим фактором в современном быстро меняющемся цифровом мире. Чем быстрее работает приложение, тем выше вероятность его успешного выполнения. Изучая алгоритмы, программисты могут выявить узкие места в коде и соответствующим образом оптимизировать его. Также помогает развить мышление, тем самым облегчает решение новых проблем, с которыми в дальнейшем придется сталкиваться. Помогает создавать более общие решения, которые можно применять к нескольким проблемам.

Изучение алгоритмов жизненно важно для любого разработчика, который хочет создавать эффективные и действенные приложения. Это помогает им решать проблемы, оптимизировать свой код для повышения производительности и выявлять закономерности и сходства между различными проблемами. Осваивая алгоритмы, программисты могут вывести

свои навыки на новый уровень и создавать инновационные, удобные и эффективные приложения.

Поэтому в качестве темы данного проекта была выбрана: «Разработка учебной онлайн-платформы с визуализацией известных алгоритмов». Она позволит студентам в условиях дистанционного обучения изучить самостоятельно указанный материал с возможностью самостоятельно написать код и сформировать отчет.

Цель проекта: работа посвящена разработке онлайн-платформы для дистанционного изучения дисциплины "Алгоритмы и анализ сложности". Система позволяет студентам изучать работу известных алгоритмов сортировки, поиска элементов, обхода дерева, вычисления контрольной суммы на основе визуализированных примеров. Также система позволяет выполнять простые упражнения в онлайн-среде с формированием отчета о проделанной работе.

Для достижения поставленной цели необходимо выполнять следующие задачи:

- разработка механизма визуализации алгоритмов;
- интеграция компилятора для работы студентов по освоению алгоритмов;
- разработка функции выгрузки отчетов по проделанной работе.

Объект исследования – онлайн-платформа для образования.

Методами исследования являются:

- анализ, анализ особенностей предметной области;
- метод прототипирования, разработка дизайна интерфейса веб-сервиса;
- метод классификации, выявление ролей пользователя и сущностей базы данных.

## ОПРЕДЕЛЕНИЯ, ОБОЗНАЧЕНИЯ, СОКРАЩЕНИЯ

Работа содержит следующие термины:

Фреймворк – программная платформа, определяющая структуру программной системы;

Компилятор – программа, переводящая написанный на языке программирования текст в набор машинных кодов;

Актор – действующий субъект;

База-данных – упорядоченный набор структурированной информации или данных;

Сервер – устройство или программное обеспечение, которое способно принимать запросы по сети и отвечать на них;

Клиент – программа, взаимодействующая с сервером;

Стек технологий – набор технологий, на основе которых разрабатывается сайт или приложение;

Контроллер – обработчик событий;

Онлайн-платформа – информационное пространство.

Работа содержит следующие сокращения:

REST – representation state transfer;

СУБД – система управления базами данных;

UML – unified modeling language;

API – application programming interface;

ORM – object-relational mapping;

ES – ECMAScript;

HTTP – hypertext transfer protocol;

SQL – structured query language;

CORS – cross-origin resource sharing;

HTML – Hypertext markup language.

## ОБЗОР ЛИТЕРАТУРЫ

При проектировании онлайн-платформы для реализации были рассмотрены средства разработки и проанализированы фреймворки, языки программирования, системы управления базами данных.

Фреймворк представляет собой платформу для разработки web-ресурсов, который имеет готовые решения и инструменты реализации различных компонентов. Был выбран язык программирования JavaScript [1] как отправная точка для клиентской и серверной части. Главными инструментами разработки в данной работе выступали Express [4] и React JS [7].

В качестве реляционной СУБД был использован PostgreSQL, с возможностью которой возможно ознакомиться в источнике [5]. Также для работы с базой данных на стороне сервера был задействован модуль Sequelize [6].

Основным инструментом для верстки служил модуль React Bootstrap и препроцессор Sass. Они позволили ускорить процесс, имея собственные особенности, ускоряющие стилизацию, подробности использования приведены в источниках [8,12].

Для обращения к серверу со стороны клиента посредством запросов был использован модуль Axios, детальнее процесс работы описан в статье [9]. Фреймворк React-router-dom [10] применялся для навигации по страницам сайта.

Требовалось генерировать отчет о проделанной работе студентом, с чем поспособствовал OfficeGen, а для сохранения – FS. В статьях подробнее описываются возможности каждой [15,16].

## **1. ЗАДАЧА ОНЛАЙН-ПЛАТФОРМЫ**

Разработанная информационная система будет предоставлять возможность просмотра пошаговой работы известных алгоритмов на визуализированном примере, на основе примера пользователю будет предоставляться возможность написания собственного кода на выбранном языке программирования из предоставляемых системой. По результату работы будет возможность выгрузить отчет в формате документа Word, в котором в дальнейшем пользователь способен вносить изменения и добавлять необходимые элементы на свое усмотрение.

### **1.1. Обзор аналогов**

С учетом задач системы найдены и проанализированы аналоги в сети Интернет: российская образовательная платформа Stepik и онлайн-платформа с алгоритмическими задачами по программированию LeetCode.

С помощью онлайн-платформы «Stepik» пользователи осваивают базовые навыки программирования на различных языках программирования, позволяет создавать собственные курсы для подготовки и развития необходимых компетенций. Тематика курсов может быть связана не только с программированием.

Благодаря платформе LeetCode появляется возможность совершенствовать навыки в различных отраслях. Присутствует градация уровней сложности задач, что помогает постепенно совершенствоваться в профессиональном плане.

Проанализировав данные сервисы, была составлена таблица 1 сравнение аналогов с требуемой информационной системой.

Таблица 1 – Сравнение критериев проанализированных сервисов и требуемого продукта

Основные критерии	Название сервиса		
	Stepik	Leetcode	Требуемый продукт
Демонстрация работы	Описываются основные принципы и пошаговые действия, но отсутствует наглядная демонстрация	Сайт переполнен различными задачами и мало где описаны принципы работы	Присутствует наглядная демонстрация работы необходимых и популярных алгоритмов
Самостоятельная работа пользователя в изучении алгоритмов	Возможность работы в компиляторе и изучение решений других пользователей	Возможность работы в компиляторе и изучение решений других пользователей	Возможность проверки результатов программы, выгрузка отчета о проделанной работе

Как видим, аналоги не предоставляют возможность выгрузить отчет о проделанной работе в формате документа Word, широкий обхват различных отраслей способен мешать в поиске необходимых материалов и их изучении. Стоит сосредоточить внимание текущего проекта на подробной визуализации и ее представлении, качестве генерируемого отчета для пользователя информационной системы.



## 2. ПРОЕКТИРОВАНИЕ СИСТЕМЫ

### 2.1. Варианты использования

В ходе анализа задач, которые информационной системе необходимо решать, были выделены 2 вида акторов: Студент и Преподаватель. Диаграмма вариантов использования представлена на рисунке 1.

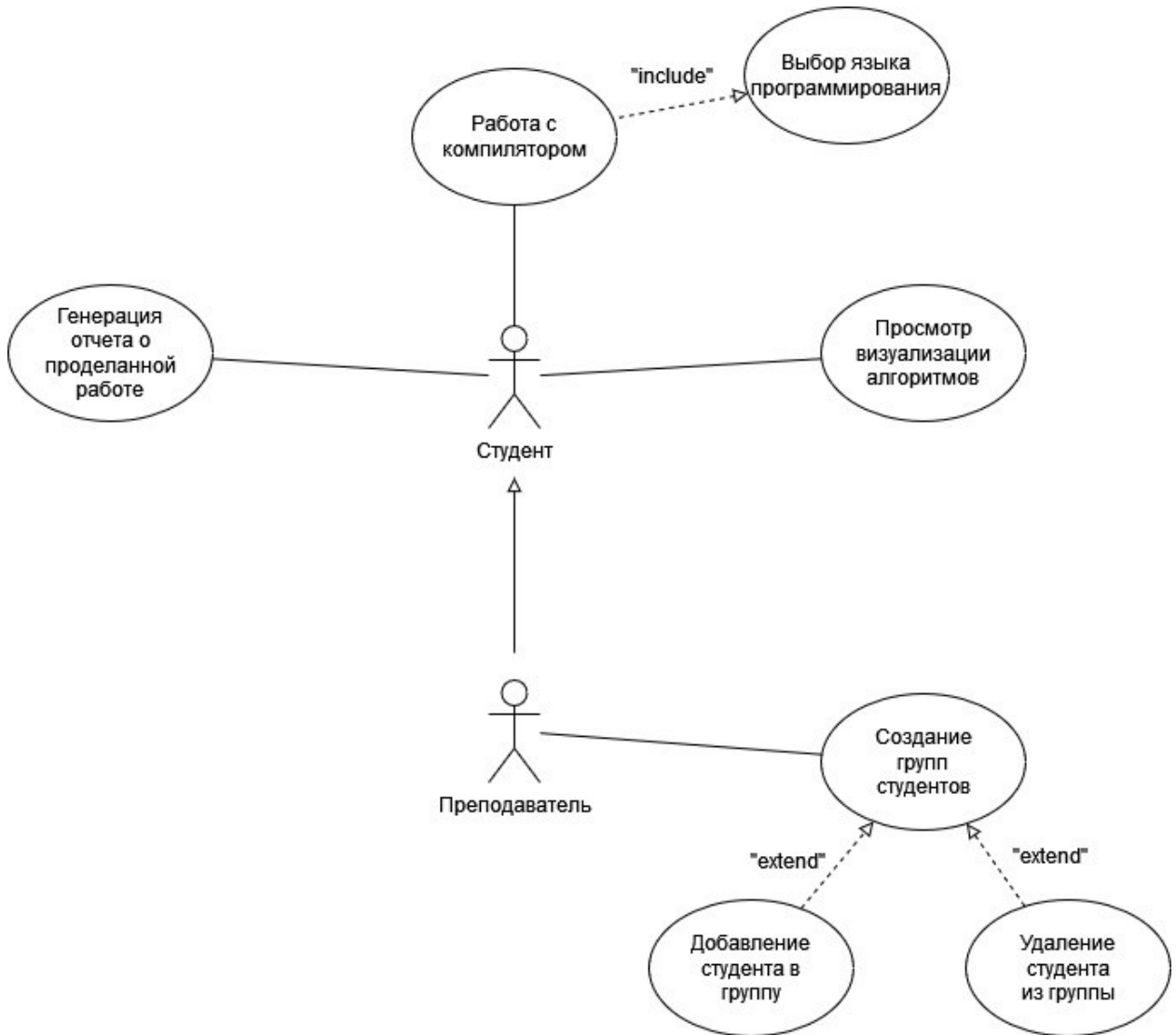


Рисунок 1 – Диаграмма вариантов использования

Далее опишем диаграммы последовательности основных функций системы к исполнению: написание кода студентом по алгоритму, выгрузка отчета о проделанной работе.

## 2.2. Диаграммы последовательности для системы

Основная работа на платформе состоит в изучении студентом алгоритма и написании собственного кода для решения поставленной задачи на языке программирования удобном для его восприятия. На рисунке 2 представлена диаграмма последовательности для поставленного варианта использования системы.

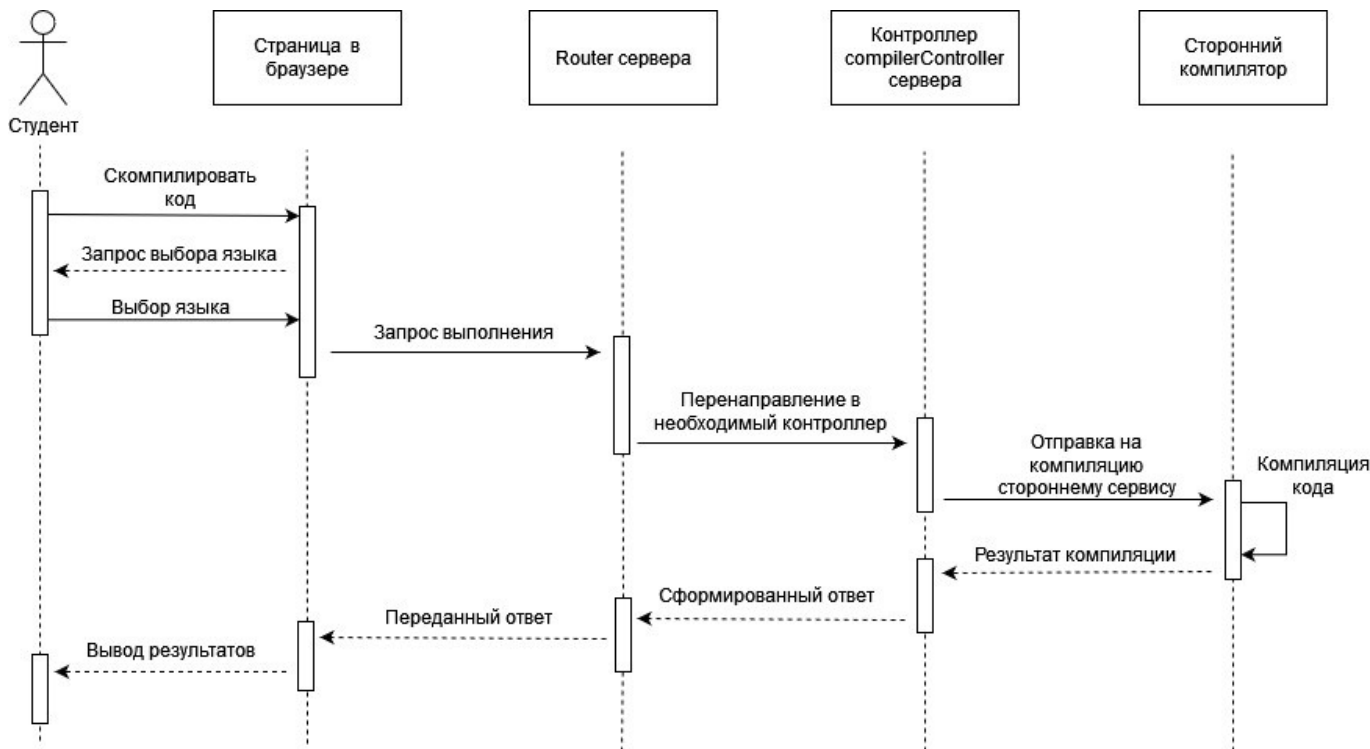


Рисунок 2 – Диаграмма последовательности для выполнения написанного кода студентом

Также немаловажной возможностью онлайн-платформы служит создание отчета о проделанной работе для последующей сдачи. Диаграмма последовательности варианта использования представлена на рисунке ниже (рис.3).

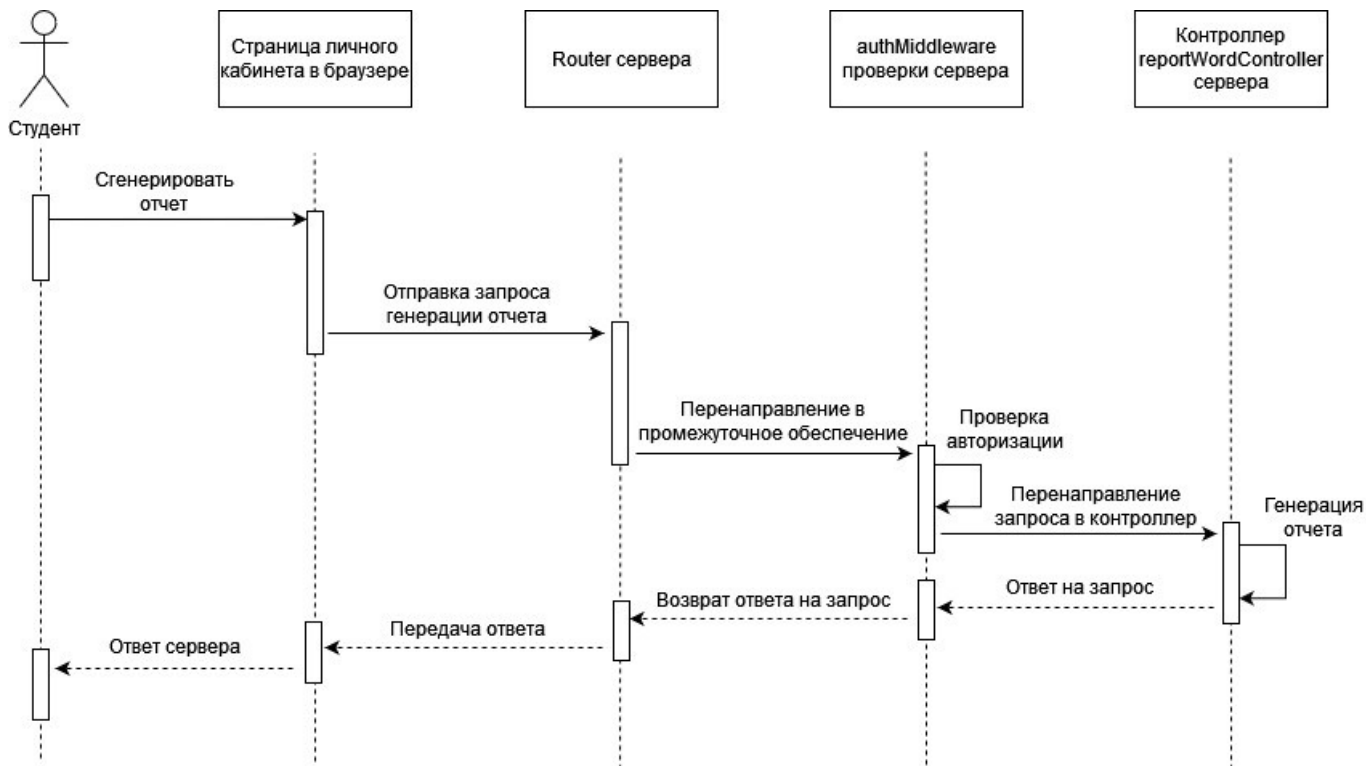


Рисунок 3 – Диаграмма последовательности для генерации отчета о проделанной работе

### 2.3. Архитектура системы

Система необходима представлять собой веб-приложение, построенное по REST архитектуре. Основное назначение REST для проектируемой системы являются такие свойства как:

- масштабируемость,
- гибкость к изменениям,
- производительность,
- отказоустойчивость,
- простота поддержки.

Построенная система будет состоять из 2 звеньев:

- Серверная часть: СУБД, REST-API сервис, который взаимодействует с СУБД и возвращающий ответ на запросы клиентской части;
- Клиентская часть: Отображение данных, вызов API методов сервера.

Клиентская часть приложения работает с серверной посредством запросов, отправляемых с помощью сети интернет через браузер. Клиентская

часть системы должна отображать различный интерфейс для каждого из определенных акторов в некоторых моментах

#### 2.4. Выбор технологий для реализации

Важным критерием для выбора был бесплатный доступ, также иным критерием являлась популярность, чтобы иметь подробную документацию и поддержку сообщества разработчиков.

Для выбора СУБД проведен анализ 3 самых распространённых реляционных СУБД. Выбор сделан в пользу реляционных, так как решение является классическим, возможность демонстрации табличных моделей для наглядности и лучшего понимания, простота установления взаимосвязи между элементами данных. Результаты сравнения популярных реляционных СУБД представлены в таблице 2.

Таблица 2 – Сравнение реляционных СУБД

СУБД	Oracle	Microsoft SQL Server	PostgreSQL
Достоинства	+ Надежность + Поддержка больших баз данных + Высокая производительность	+ Быстродействие + Документация + Бесплатная версия	+ Высокая производительность + Бесплатная + Документация + Поддержка больших баз данных
Недостатки	– Цена – Ресурсозатратность	– Цена – Ресурсозатратность	– Задержки при выполнении операций

Oracle подойдет для крупных организаций, которые работает с большим объемом данных, Microsoft SQL Server по достоинствам хорошо подходит для разработки информационных систем небольших и средних. Выбор сделан в сторону PostgreSQL в виду того, что он бесплатный, обеспечивая

необходимую производительность, присутствует большой объем документации.

Стек технологий для реализации проекта был взят для языка программирования JavaScript по причине того, что он является одним из самых популярных языков программирования в мире, который используется для создания интерактивных веб-сайтов, приложений и игр. Легок в изучении и использовании, благодаря своей простоте и доступности. Также стоит отметить то, что не придется одновременно работать на двух языках программирования, так как основных исполнителей разработки данной системы – один.

Причины сделанного выбора в пользу JavaScript:

1) Популярность. Используется повсеместно, поэтому имеется множество ресурсов и сообществ, где возможно получить поддержку и помощь.

2) Всесторонность. Может использоваться как на стороне клиента (для создания интерактивных веб-сайтов), так и на стороне сервера (для создания веб-приложений).

3) Поддержка. Многие браузеры поддерживают JavaScript, что позволяет создавать кросс-браузерные приложения.

4) Фреймворки. Существует многие фреймворки и библиотеки, которые в свою очередь упрощают и ускоряют процесс создания приложения.

5) Динамическая типизация. Позволяет работать с переменными разного типа, что делает его более гибким и удобным в использовании.

6) Обучение. Доступен для изучения как новичкам, так и опытным разработчикам, благодаря широкому спектру полезных ресурсов и сообществ.

Преимущества JavaScript в сравнении с иными языками программирования:

– Java. Схожий синтаксис, но используют различные парадигмы программирования. Обычно Java является более строгим и менее гибким по сравнению с JavaScript.

– Python. Python и JavaScript оба являются интерпретируемыми языками программирования, но Python используется для создания широкого спектра приложений, включая веб-приложения, машинное обучение и научные вычисления. Обычно Python считается более простым и более читаемым языком, чем JavaScript.

– C++. Являются разными языками программирования, использующими разные парадигмы программирования (C++ использует процедурную и объектно-ориентированную парадигмы). C++ используется для создания системного программного обеспечения, игр и программного обеспечения для встроенных систем. Обычно C++ является более быстрым и эффективным языком, чем JavaScript, но требует большего количества кода для достижения того же результата.

– PHP. оба используются для создания веб-приложений, но PHP обычно используется для создания серверных приложений, тогда как JavaScript используется для создания клиентских приложений. PHP является более строгим и менее гибким языком по сравнению с JavaScript, но имеет ряд плюсов, включая более простую работу с базами данных и удобство работы с файлами.

– C#. Имеет более строгую типизацию, что делает его более надежным и безопасным в отношении ошибок во время выполнения программы. Подходит для создания крупных приложений, благодаря мощной стандартной библиотеке и возможности использования объектно-ориентированного подхода. Но если требуется создать динамический пользовательский интерфейс для веб-страницы, то JavaScript может быть лучшим выбором

Используемые технологии для реализации онлайн-платформы для изучения алгоритмов приведены ниже. За основу взят фреймворк Express, который является основным и самым распространённым для платформы

NodeJS, реализующих серверную часть крупных порталов в сети Интернет на данный момент.

Стек технологий для BackEnd:

- NodeJS, платформа для создания сервера на JavaScript,
- Express, для написания серверной части на NodeJS,
- PostgreSQL, система управления базами данных,
- Sequelize, ORM для реляционных баз данных на NodeJS,
- Officegen, формирование документа Word на сервере,
- FS, модуль для работы с файловой системой.

Выбор фреймворка для клиентской части приложения на языке JavaScript уже более обширный. Проведен анализ трех наиболее распространенных для решения данной задачи. Результаты сравнения представлены в таблице 3.

Таблица 3 – Сравнение фреймворков для реализации клиентской части приложения

Фреймворк	React	Angular	Vue
Достоинства	+ Гибкость в создании компонентов + Высокая производительность + Обширное сообщество и готовых библиотек	+ Начальная интеграция необходимого стека + Понятный шаблонизатор + Понятная структура проекта	+ Легок в освоении + Небольшой размер + Простой шаблонизатор
Недостатки	– Неудобный шаблонизатор	– Высокий порог входа – Большой размер	– Не такое большое сообщество – Меньшее количество готовых библиотек

Angular является отличным решением для крупных компаний, ввиду понимания структуры проекта при его масштабировании. Vue легок в освоении и изучении, но не так популярен как первые два варианта. React отлично подходит для необходимой разработки по причинам того, что присутствует большое сообщество и готовые решения, также стоит отметить производительность, которая потребуется для информационной системы.

Стек технологий для FrontEnd:

- React JS,
- React Bootstrap, для верстки элементов, графики,
- Axios, запросы к серверу,
- React-router-dom, навигация на сайте,



- Mobx, стейт-менеджер,
- Sass, препроцесс для ускорения стилизации,
- React-Vis, библиотека для построения диаграмм и графов, разработанная Uber.

В качестве сборщика модулей клиентской части приложения на React JS был выбран Vite по нескольким причинам: быстрая скорость сборки из-за использования технология ES modules и HTTP/2, простота использования через интерфейс командной строки, ускоряющая начало работы отличная документация, разработка в реальном времени.

### 3. ПРОГРАММНАЯ РЕАЛИЗАЦИЯ ОНЛАЙН-ПЛАТФОРМЫ

#### 3.1. Разработка базы данных информационной системы

Для создания базы данных необходимо определиться с необходимыми сущностями, какую информацию должна хранить. В результате анализа были выделены 6 сущностей

- 1) Пользователь. Хранит в себе информацию о пользователях.
- 2) Группа. Сущность хранит в себе информацию группе.
- 3) Университет. Основная информация о университетах.
- 4) Код. Таблица содержит скрипт, с которым работал пользователь.
- 5) Скомпилированный Код. Таблица для хранения статусов работы над алгоритмом.
- 6) Поделиться кодом. Таблица, хранящая информацию об отправке работы студентом.

Была разработана логическая модель данных для проверки «валидности» будущей базы данных и связи сущностей (рис.4).

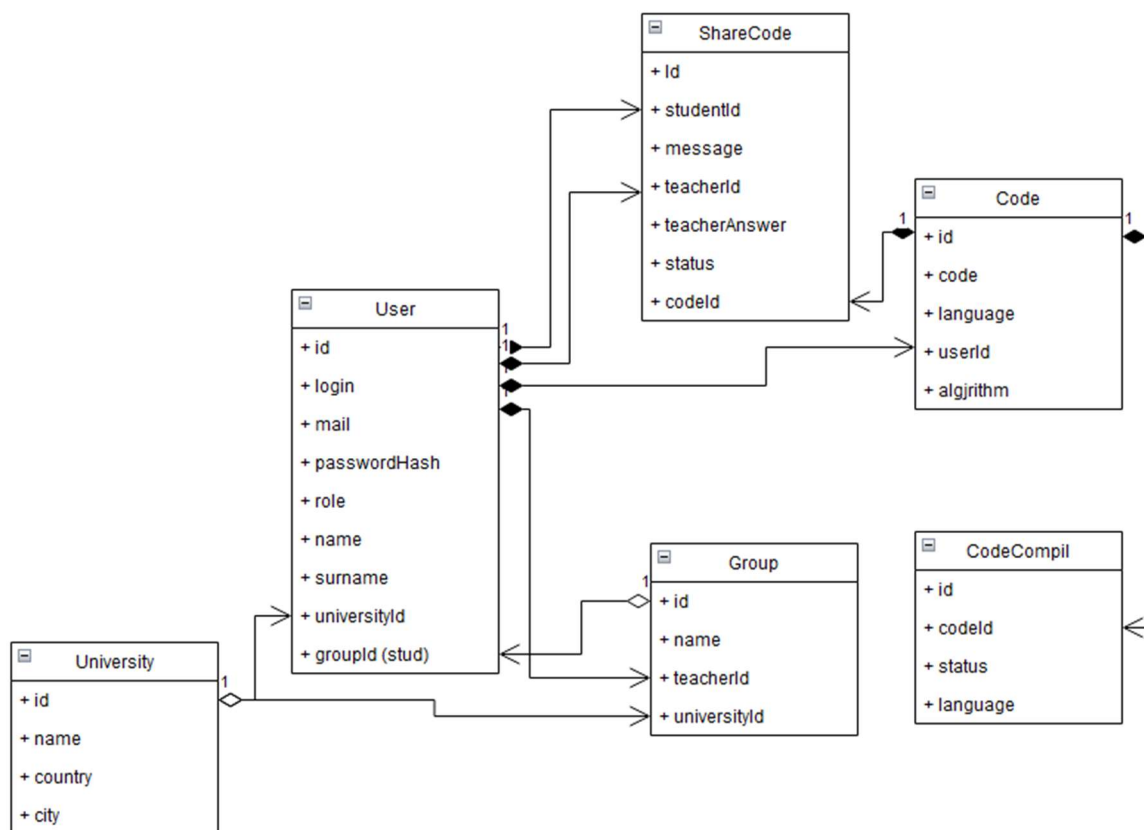


Рисунок 4 – Логическая модель базы данных

На основе логической модели была создана база данных в СУБД PostgreSQL с использованием технологии Sequelize для NodeJS с подходом проектирования code-first, написаны модели таблиц и связей между ними, по которым сгенерирована база данных. Результат представлен на рисунке 5.

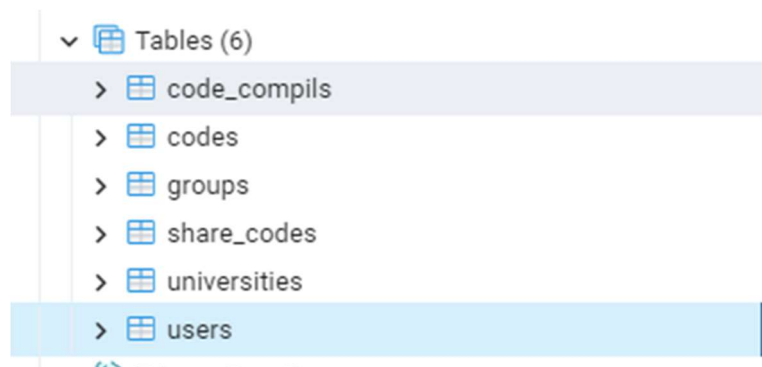


Рисунок 5 – Сгенерированные таблицы базы данных

В дальнейшем для работы используются написанные модели таблиц для запросов на получение, редактирование, добавления и удаление записей, что упрощает процесс, необязательно писать SQL запросы к базе данных в формате строки, а приставлены вложенным кодом для упрощения понимания благодаря технологии Sequelize.

### 3.2. Разработка серверной части приложения

Клиентская и серверная часть приложения разделены и связаны API для взаимодействия. Для реализации серверной части был использован модуль Express под платформу NodeJS, разбит на части для облегчения дальнейшей работы, расширения и структурирования программы. Разбиение представлено в виде UML диаграммы пакетов для понимания (рис.6).

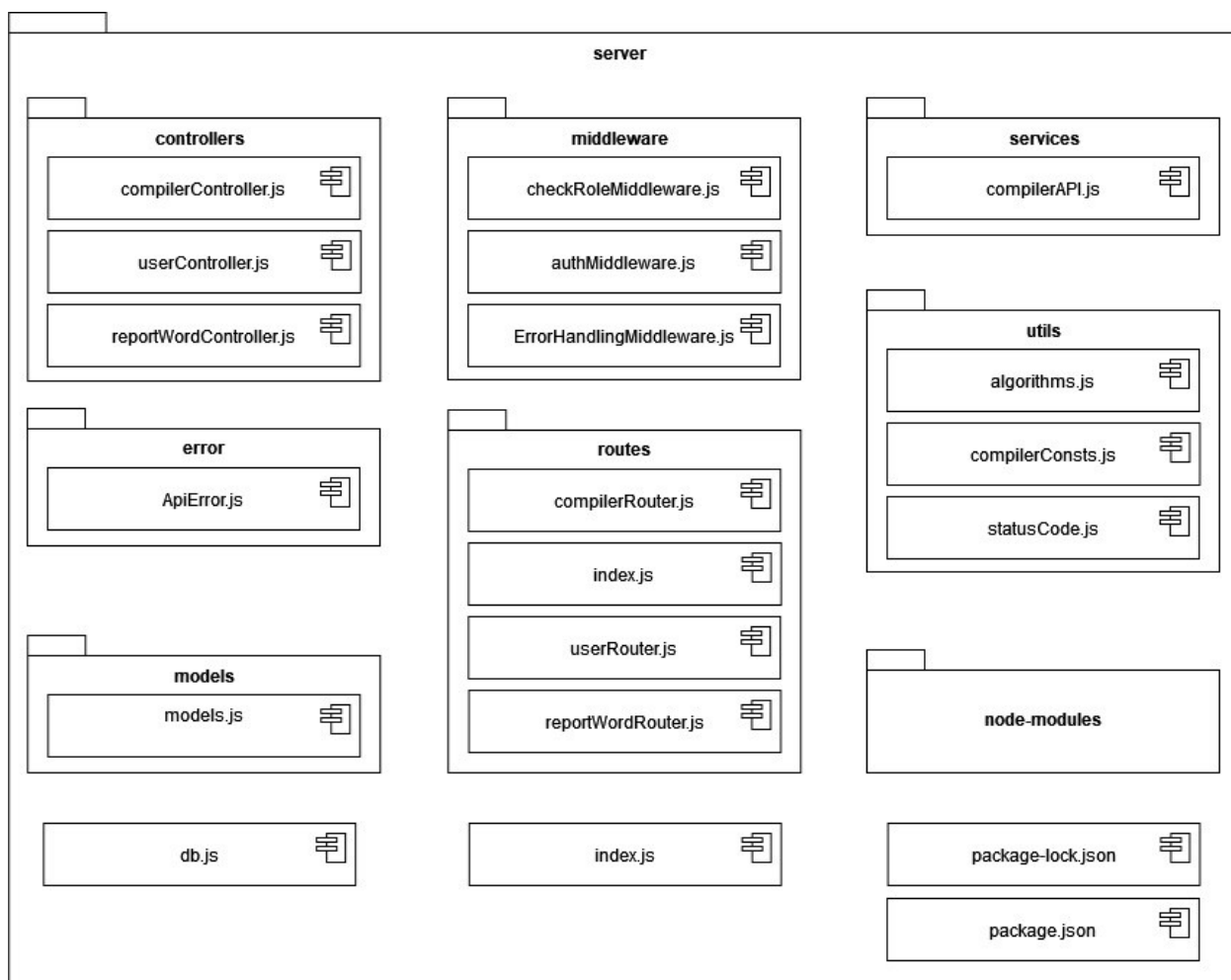


Рисунок 6 – UML диаграмма пакетов структуры серверной части

Для того, чтобы передавать запрос к серверу в необходимый метод контроллера через промежуточные слои, был задействован модуль Router и описаны пути его взаимодействия. Такие пути разделены на 3 категории: работы компилятора, с пользователями, генерации отчетов.

Также для получения самого запроса необходимо было прослушивать установленный порт используя стандарт CORS для доступа к объектам

сторонних интернет-ресурсов, данная задача решена с помощью Cors, которая просто интегрируется в решения на Express.

Некоторые из запросов требуют, чтобы пользователь был авторизован или присутствовали необходимые права, то есть возникает нужда в промежуточных слоях для обеспечения обработки данных вопросов.

Были разработаны три Middleware-обработчика для проблемы, описанной выше:

- 1) `authMiddleware`, проверка на то, авторизован ли пользователь. Проверяет наличие и валидность ключа из «головы» запроса и декодирует его;
- 2) `checkMiddleware`, проверка роли пользователя, сравнивает права роли пользователя с необходимыми;
- 3) `ErrorHandlingMiddleware`, проверка и финальная обработка запроса при возникновении ошибки.

Для обработки ошибок был разработан класс `ApiError`, который формирует ответ на запрос при вызове. В составе запроса с ошибкой указывается код ошибки и сообщение, включающее причину возникновения.

Использование изображений на сайте влечет за собой ее хранения, на стороне клиентской части влечет за собой проблемы с дальнейшей работой, для решения изображения раздаются как статика и хранятся отдельно с возможностью доступа через браузер по ссылке. Также данный подход походит для доступа к сгенерированным отчетам, они раздаются аналогичным образом.

Разработано три контроллера для обработки запросов пользователя:

- a. `UserController` для логики связанной с пользователем;
- b. `compilerController` для логики связанной с компилятором;
- c. `reportWordController` для генерации отчета в формате документа Word.

Написано три метода для работы с пользователями: вход, регистрация и аутентификация. Каждый из них работает с токенами, которые отправляют в

виде ответа. Использовался `json-web-token` для генерации самого токена, время жизни которого было указано 12 часов.

Хранение пароля в исходном в состоянии не допустимо, требуется его хешировать для безопасности утечки информации и пользователей, с данной задачей связано использование `bcrypt`, который шифрует пароль необходимое количество раз (выбрано 5). При входе пользователя берется пароль из запроса и сравнивается с паролем из базы данных с помощью метода `bcrypt.compareSync`.

Также в контроллере для обработки запросов связанного с пользователями имеются методы для работы с группами студентов: создание групп, добавление студентов в группу и их удаления из нее.

Обработка запросов компиляции кода происходит в контроллере `compilerController`, где в начале полученного скрипта вставляется начальная часть с необходимыми объявлениями относительно выбранного алгоритма и языка программирования. Ответ состоит из полей: выходные данные, статус код, необходимой памяти для скрипта и времени выполнения. Также ошибки при ее возникновении при компиляции.

Присутствует и метод для сохранения кода написанного пользователем в базу данных с указанием конкретного алгоритма, для которого писался код, и языком для него.

Для генерации отчетов в Word присутствует контроллер `reportWordController`, он использует модули `OfficeGen` и `FS`. `OfficeGen` для создания и генерации самого отчета Word, в который записывается необходимая информация о проделанной работе. Требуется как-то передавать отчет на сторону клиента, для решения данной задачи был задействован модуль `FS` для работы с файловой системой, он сохраняет подготовленный отчет в необходимой папке для последующей раздачи его как статику и скачивания.

### 3.3. Тестирование серверной части приложения

Тестирование сервера производилось с использованием программы Postman, инструмент для работы с API, который позволяет посылать запросы к сервису и работать с их ответами. Выбран по следующим причинам:

- 1) интуитивно-понятен и прост в использовании;
- 2) не требует никакой лицензии;
- 3) поддержка разных API;
- 4) запускается на любых операционных системах;
- 5) поддерживает ручное и автоматическое тестирование.

Пример проверки входа с использованием Postman представлен на рисунке 7, также компиляции на рисунке 8.

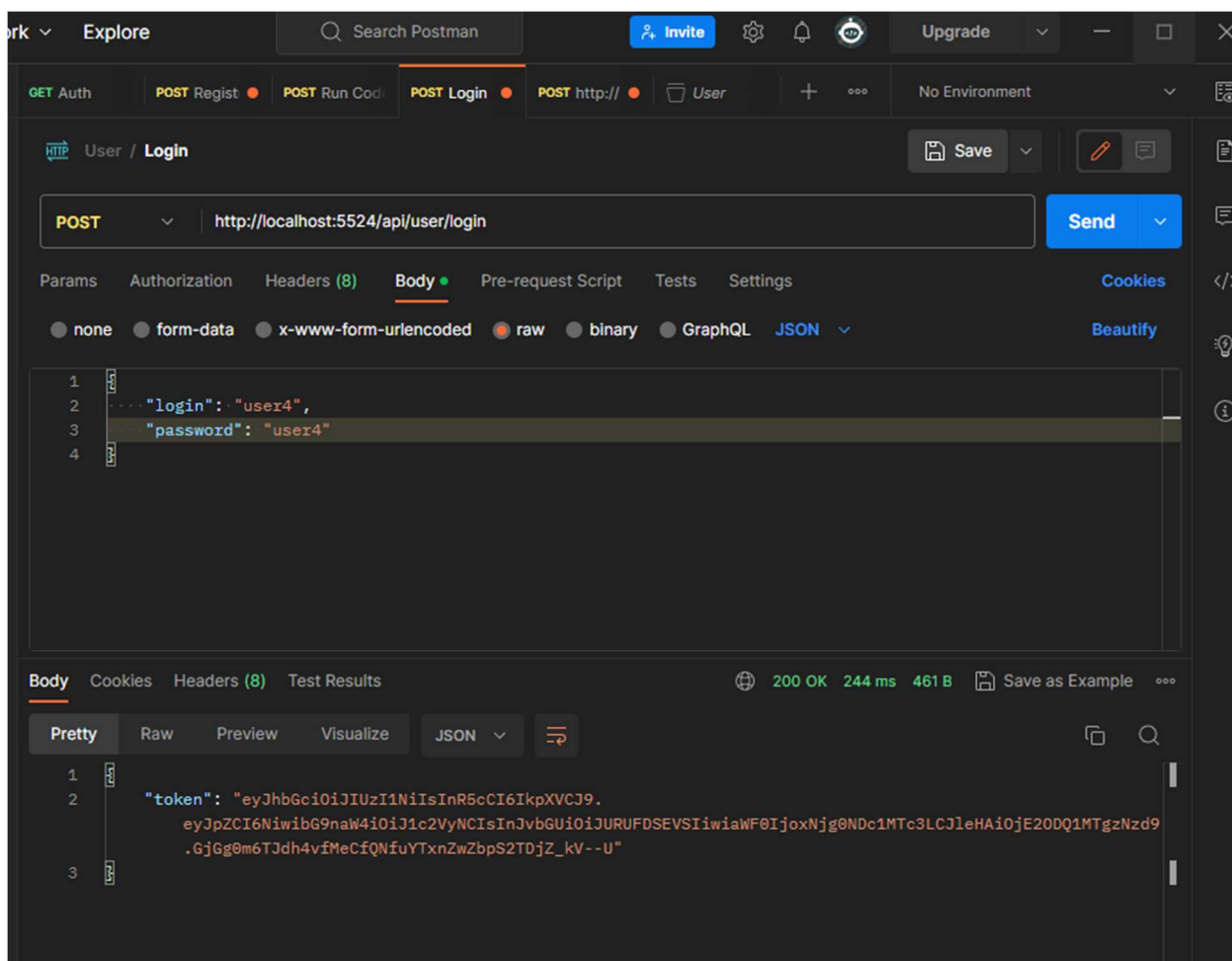


Рисунок 7 – Тестирование запроса Login

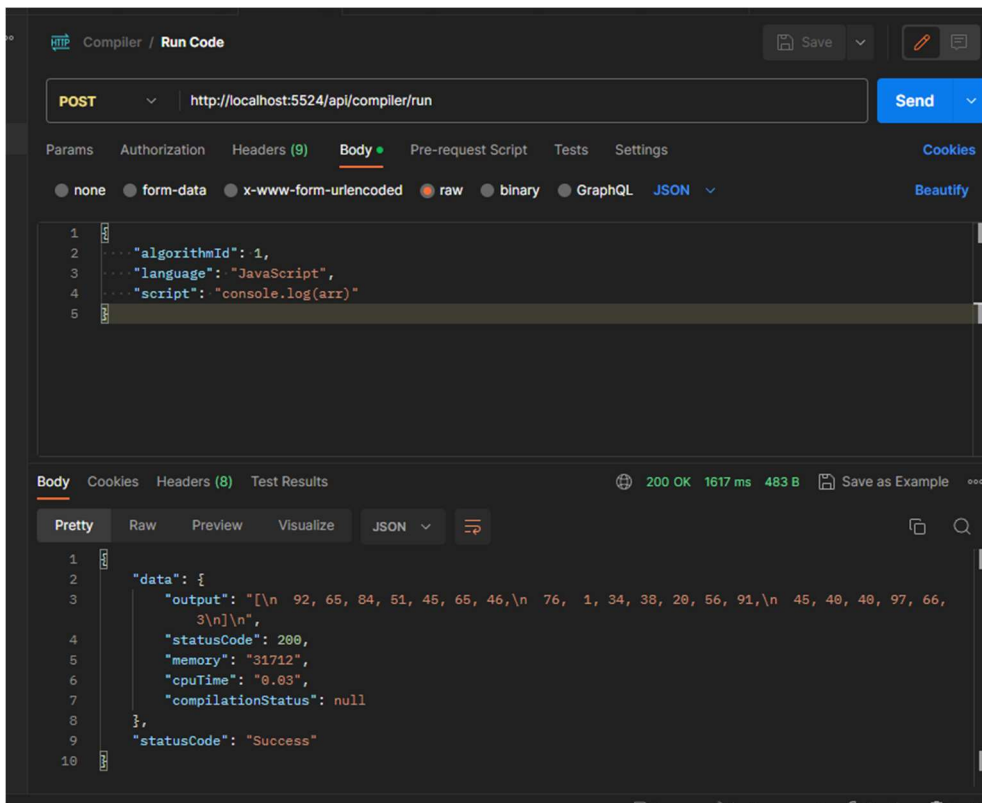


Рисунок 8 – Тестирование запроса Run Code

Также демонстрация запроса на генерацию отчета Word представлена на рисунке 9.

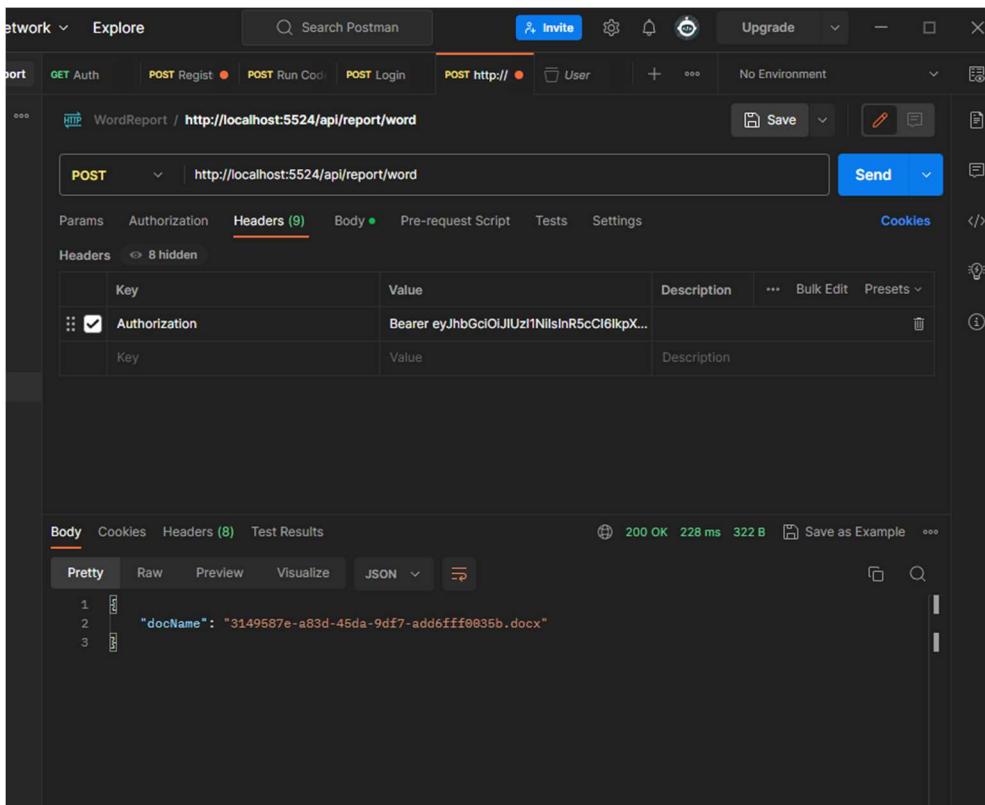


Рисунок 9 – Тестирование запроса генерации отчета



### 3.4. Разработка клиентской части приложения

React JS фреймворк был использован вместе со сборщиком Vite. По результату работы была составлена диаграмма компонентов информационной системы в языке моделирования UML. Результат представлен на рисунке ниже (рис.10)

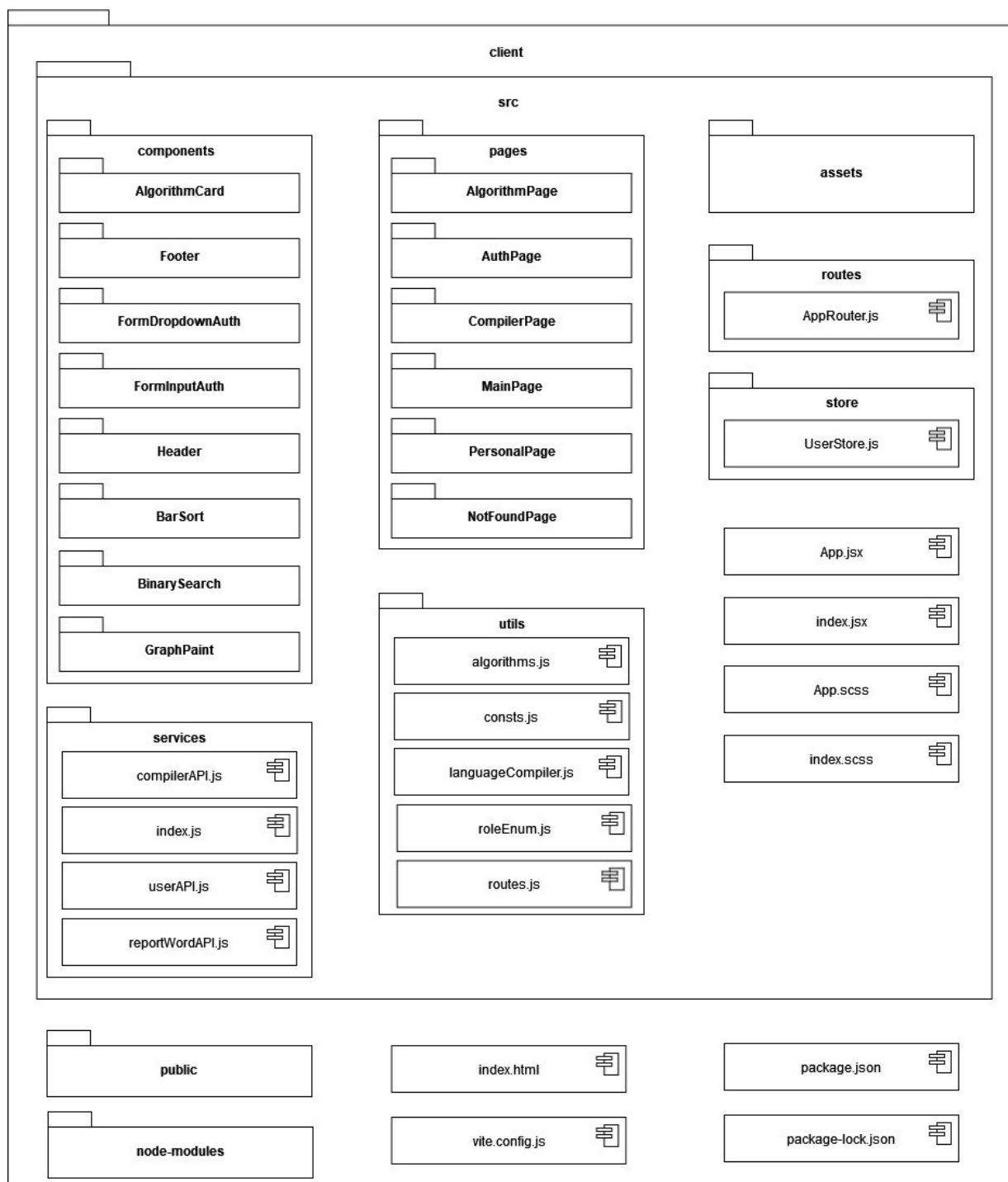


Рисунок 10 – UML диаграмма пакетов структуры клиентской части

Были описаны пути адресов сайта и к какому компоненту они должны вести, для данной и дальнейшей навигации была использована библиотека React-Router-Dom с помощью компонентов Routes, Route, useNavigate.

Также присутствуют сервисы, страницы, хранилище и компоненты. В хранилище присутствует основная информация о пользователе и авторизован ли он.

В папке с сервисами описаны методы для отправки запросов на сервер и часть обработки ответов от них. Например, для работы логина, аутентификации и регистрации пользователя отправляют по необходимому адресу запрос с его телом, получая токен и сохраняя его в локальное хранилище браузера, возвращая при этом декодированную информацию о пользователе.

Все запросы описаны с использованием библиотеки Axios, основан на промисах, упрощая создание запросов к заданной конечной точке. Легко задавать тело и голову запроса, указывая необходимые параметры как Authorization, responseType и так далее.

Были разработаны основные, часто используемые компоненты, которые используются на страницах: карточка алгоритма, «хедер» сайта, «футер» сайта, поле ввода и всплывающий список.

Реализована главная страница, на которой будут представлены алгоритмы (рис.11).



Рисунок 11 – Главная страница

Для входа и регистрации используется одна страница, которая перестраивается в зависимости от адреса в поисковой строке, полученной через `useLocation`. Пример представлен рисунках ниже 12-13, для реализации были использованы компоненты из библиотеки `React-Bootstrap`.

## Авторизация

Введите login или email

Введите пароль

Нет аккаунта? [Зарегистрироваться](#)

Рисунок 12 – Окно авторизации

## Регистрация

Введите login

Введите пароль

Введите почту

Введите роль

Введите имя

Введите фамилию

Есть аккаунт? [Войдите](#)

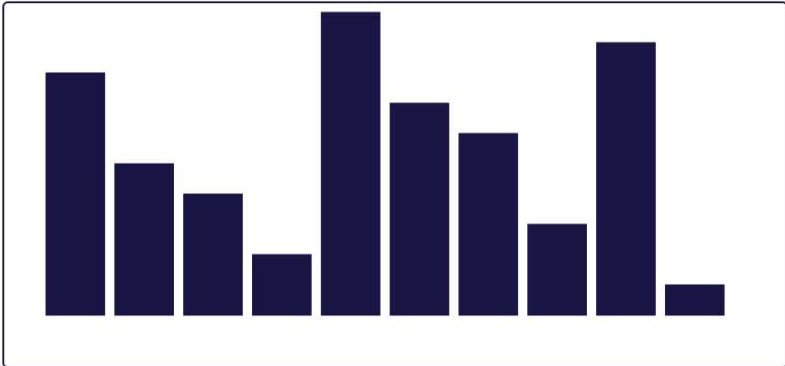
Рисунок 13 – Окно регистрации

Одно из основных окон – это визуализация алгоритма. Само по себе представление является либо столбчатой диаграммой, либо нарисованной с помощью Canvas графики для наглядности процесса. Справа представлен пример кода с клавишами «Компилятор» для самостоятельного написания программы и «Демонстрация» для представления сортировки.

При переходе на данную страницу имеем на представлении исходные данные (рис.14). При нажатии на вторую клавишу функция обрабатывает массив элементов, в примере кода подсвечивается выполняемая строка, а на диаграмме элементы, которые сравниваются (рис.15). Если необходимо, то значения меняются.

Algorithm Platform Кабинет Выход

### Пузырьковая сортировка



```
function bubbleSort(arr) {
  for (let j = arr.length - 1; j > 0; j--) {
    for (let i = 0; i < j; i++) {
      if (arr[i] > arr[i + 1]) {
        [arr[i], arr[i+1]] = [arr[i+1], arr[i]];
      }
    }
  }
}
```

Компилятор Демонстрация

#### Описание

Самый простой, но не самый эффективный алгоритм. Эффективность работы алгоритма -  $O(n^2)$

#### Принцип работы

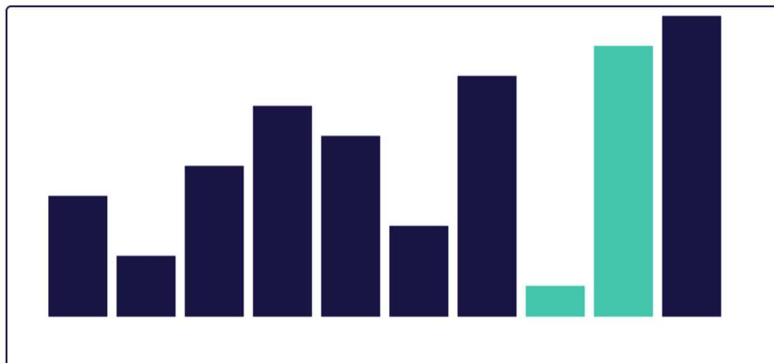
На каждом шаге мы находим наибольший элемент из двух соседних и ставим этот элемент в конец пары. Получается, что при каждом прогоне цикла большие элементы будут всплывать к концу массива, как пузырьки воздуха — отсюда и название.

#### Алгоритм выглядит так:

1. Берём самый первый элемент массива и сравниваем его со вторым. Если первый больше второго — меняем их местами с первым, если нет — ничего не

Рисунок 14 – Стартовое представление визуализации

## Пузырьковая сортировка



```
function bubbleSort(arr) {
  for (let j = arr.length - 1; j > 0; j--) {
    for (let i = 0; i < j; i++) {
      if (arr[i] > arr[i + 1]) {
        [arr[i], arr[i+1]] = [arr[i+1], arr[i]];
      }
    }
  }
}
```

Компилятор

Демонстрация

### Описание

Самый простой, но не самый эффективный алгоритм. Эффективность работы алгоритма -  $O(n^2)$

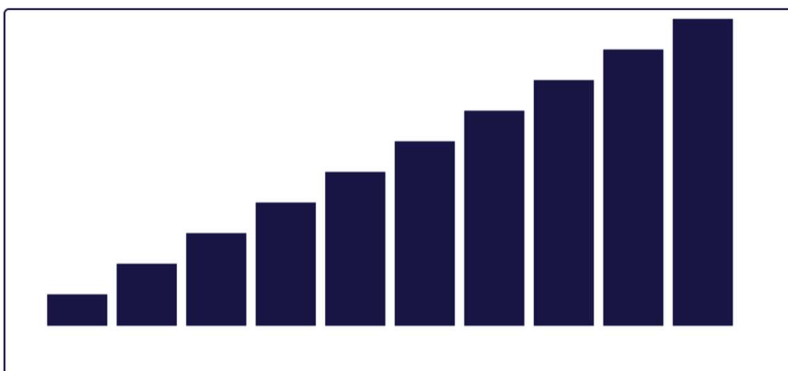
### Принцип работы

На каждом шаге мы находим наибольший элемент из двух соседних и ставим этот элемент в конец пары. Получается, что при каждом прогоне цикла большие элементы будут всплывать к концу массива, как пузырьки воздуха — отсюда и название.

Рисунок 15 – Визуализация работы алгоритма

По окончании демонстрируется итоговый результат выполнения программы представлен в виде диаграммы (рис.16).

## Пузырьковая сортировка



```
function bubbleSort(arr) {
  for (let j = arr.length - 1; j > 0; j--) {
    for (let i = 0; i < j; i++) {
      if (arr[i] > arr[i + 1]) {
        [arr[i], arr[i+1]] = [arr[i+1], arr[i]];
      }
    }
  }
}
```

Компилятор

Демонстрация

### Описание

Самый простой, но не самый эффективный алгоритм. Эффективность работы алгоритма -  $O(n^2)$

### Принцип работы

На каждом шаге мы находим наибольший элемент из двух соседних и ставим этот элемент в конец пары. Получается, что при каждом прогоне цикла большие элементы будут всплывать к концу массива, как пузырьки воздуха — отсюда и название.

### Алгоритм выглядит так:

Рисунок 16 – Результат выполнения алгоритма

Для визуализации алгоритмов сортировки были выполнены с использованием библиотеки React-Vis, разработанной компанией Uber, остальные с помощью Canvas. В общей сложности было визуализировано 5 алгоритмов:

- 1) пузырьковая сортировка,
- 2) сортировка расчёской,
- 3) быстрая сортировка,
- 4) бинарный поиск,
- 5) обход графа в ширину.

Пример, визуализации алгоритма сортировки расчёской представлен на рисунках 17-19.

Algorithm Platform
Кабинет
Выход

### Сортировка расчёской

```
function combSort(arr) {
  const factor = 1.247;
  let gapFactor = arr.length / factor;
  while (gapFactor > 1) {
    const gap = Math.round(gapFactor);
    for (let i = 0, j = gap; j < arr.length; i++, j++) {
      if (arr[i] > arr[j]) {
        [arr[i], arr[j]] = [arr[j], arr[i]];
      }
    }
    gapFactor = gapFactor / factor;
  }
}
```

**Описание**  
 Улучшаем пузырьковую сортировку. Худшее время -  $O(n^2)$ , лучшее время -  $O(n \cdot \log n)$   
 В чём хитрость сортировки расчёской

Компилятор
Демонстрация

Рисунок 17 – Начальное состояние сортировки расчёской

Algorithm Platform
Кабинет
Выход

### Сортировка расчёской

```
function combSort(arr) {
  const factor = 1.247;
  let gapFactor = arr.length / factor;
  while (gapFactor > 1) {
    const gap = Math.round(gapFactor);
    for (let i = 0, j = gap; j < arr.length; i++, j++) {
      if (arr[i] > arr[j]) {
        [arr[i], arr[j]] = [arr[j], arr[i]];
      }
    }
    gapFactor = gapFactor / factor;
  }
}
```

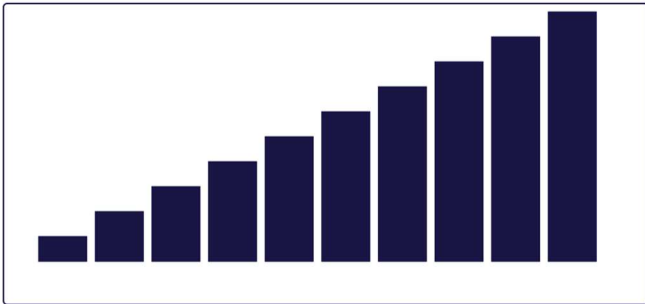
**Описание**  
 Улучшаем пузырьковую сортировку. Худшее время -  $O(n^2)$ , лучшее время -  $O(n \cdot \log n)$   
 В чём хитрость сортировки расчёской

Компилятор
Демонстрация

Рисунок 18 – Промежуточное состояние сортировки расческой

Algorithm Platform Кабинет Выход

### Сортировка расчёской



```
function combSort(arr) {
  const factor = 1.247;
  let gapFactor = arr.length / factor;
  while (gapFactor > 1) {
    const gap = Math.round(gapFactor);
    for (let i = 0, j = gap; j < arr.length; i++, j++) {
      if (arr[i] > arr[j]) {
        [arr[i], arr[j]] = [arr[j], arr[i]];
      }
    }
    gapFactor = gapFactor / factor;
  }
}
```

**Описание**  
Улучшаем пузырьковую сортировку. Худшее время -  $O(n^2)$ , лучшее время -  $O(n \cdot \log n)$   
В чём хитрость сортировки расчёской

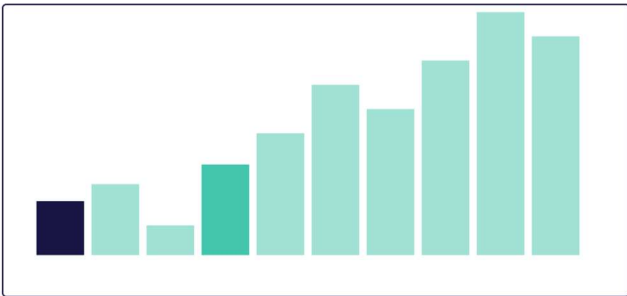
Компилятор Демонстрация

Рисунок 19 – Финальное состояние сортировки расчёской

Пример, промежуточного состояния быстрой сортировки представлен на рисунке 20, стартовое и финальное состояние идентично прошлым алгоритмам. Опорный элемент выделен темно-мятным цветом.

Algorithm Platform Кабинет Выход

### Быстрая сортировка



```
function quickSort(arr) {
  if (arr.length < 2) return arr;
  let pivot = arr[0];
  const left = [];
  const right = [];
  for (let i = 1; i < arr.length; i++) {
    if (pivot > arr[i]) {
      left.push(arr[i]);
    } else {
      right.push(arr[i]);
    }
  }
  return quickSort(left).concat(pivot, quickSort(right));
}
```

**Описание**  
Ей уже 60 лет, но она до сих пор работает быстро. В среднем сложность составляет  $O(n \cdot \log n)$ .  
В чём идея быстрой сортировки  
Взята за основу классическая пузырьковая сортировка и преобразована так: 1. На первом шаге выбирается опорный элемент (шарик) и выделен

Компилятор Демонстрация

Рисунок 20 – Промежуточное состояние быстрой сортировки

Для визуализации бинарного поиска был использован встроенный HTML элемент Canvas. Для бинарного поиска берется элемент в центре отсортированного массива, поиск происходит числа 75. Для демонстрации обхода в ширину графа также задействован аналогичный элемент. Пример, их работы представлен на рисунках 21-23.



## Бинарный поиск

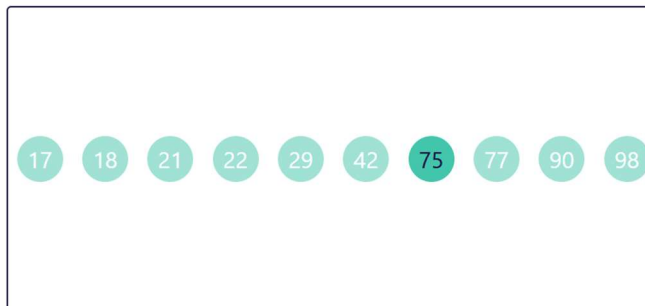


### Описание

Рисунок 21 – Промежуточное состояние бинарного поиска

```
function binarySearch(arr, key){
  let start = 0;
  let end = arr.length - 1;
  while (start <= end) {
    let middle = Math.floor((start + end) / 2);
    if (sortedArray[middle] === key) {
      return middle;
    } else if (sortedArray[middle] < key) {
      start = middle + 1;
    } else {
      end = middle - 1;
    }
  }
}
```

## Бинарный поиск



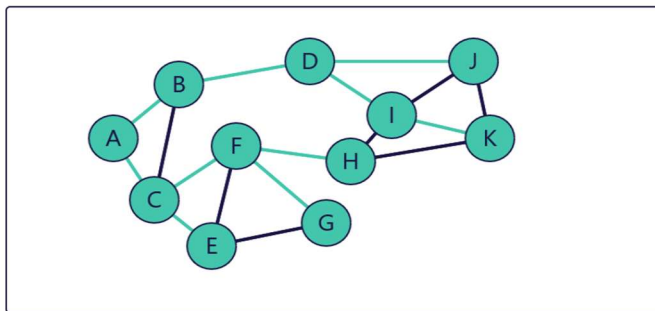
### Описание

Классический алгоритм поиска элемента в отсортированном массиве (векторе), использующий дробление массива на половины. Используется в информатике,

```
function binarySearch(arr, key){
  let start = 0;
  let end = arr.length - 1;
  while (start <= end) {
    let middle = Math.floor((start + end) / 2);
    if (sortedArray[middle] === key) {
      return middle;
    } else if (sortedArray[middle] < key) {
      start = middle + 1;
    } else {
      end = middle - 1;
    }
  }
  return -1;
}
```

Рисунок 22 – Финальное состояние бинарного поиска

## Обход в ширину



### Описание

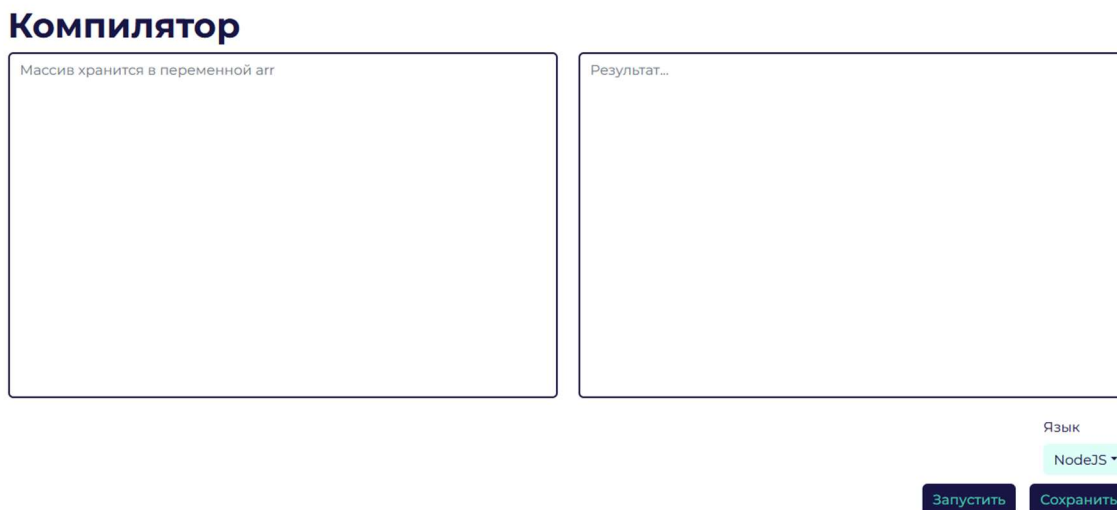
Один из простейших алгоритмов обхода графа, являющийся основой для многих важных алгоритмов для работы с графами.

Пусть задан невзвешенный ориентированный граф  $G=(V,E)$ , в котором выделена

```
function bfs(adj, st, to) {
  let queue = [];
  queue.push(st);
  st.visited = true;
  while(queue.length > 0) {
    let v = queue.shift();
    for(let neighbor of adj[v]) {
      if(!neighbor.visited) {
        queue.push(neighbor);
        neighbor.visited = true;
      }
      if(neighbor === to) return true;
    }
  }
  return false;
}
```

Рисунок 23 – Финальное состояние обхода в ширину

При нажатии на кнопку «Компилятор» происходит переход на другую страницу, где появляется возможность самостоятельно написать код с возможностью выбора языка программирования (рис.24). В левом окне предоставляется писать скрипт, в правом его результат выполнения. Снизу сохраняется описание алгоритма.

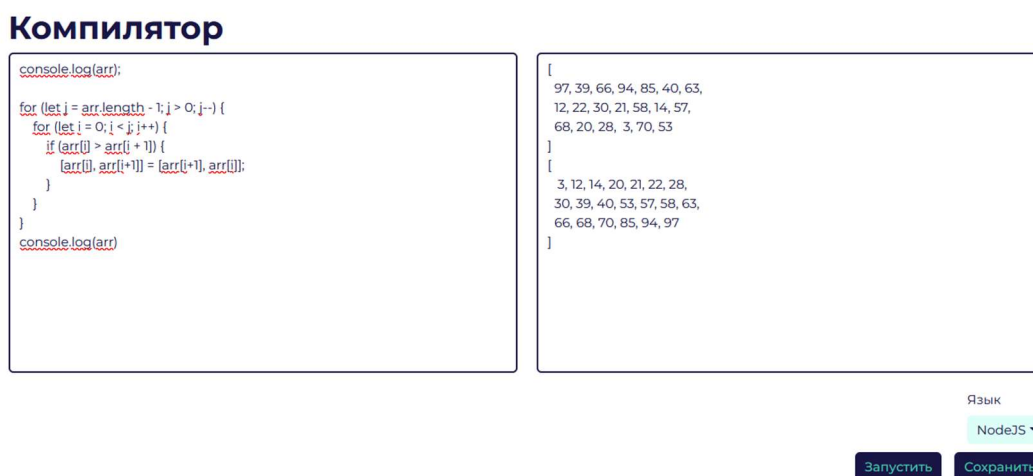


#### Описание алгоритма

Классический алгоритм поиска элемента в отсортированном массиве (векторе), использующий дробление массива на половины. Используется в

Рисунок 24 – Окно компилятора

Когда пользователь выбрал язык и написал скрипт, появляется возможность проверить скрипт на работоспособность. Результат ответа на запрос отображается в правом окне (рис.25).

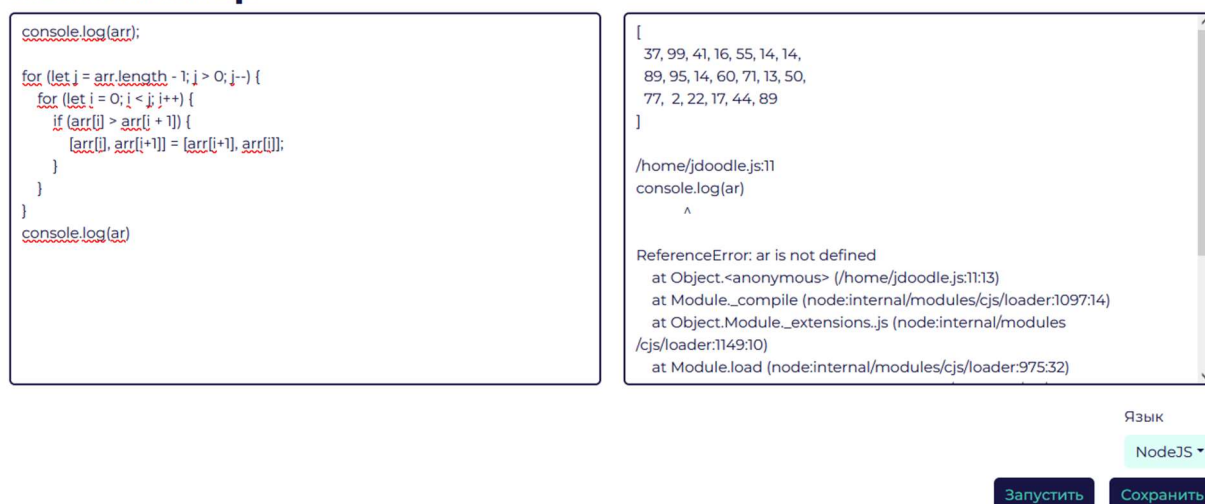


#### Описание алгоритма

Рисунок 25 – Проверка работы компилятора

Также при некорректности написанного скрипта, информация об ошибке будет находиться в правой части (рис.26).

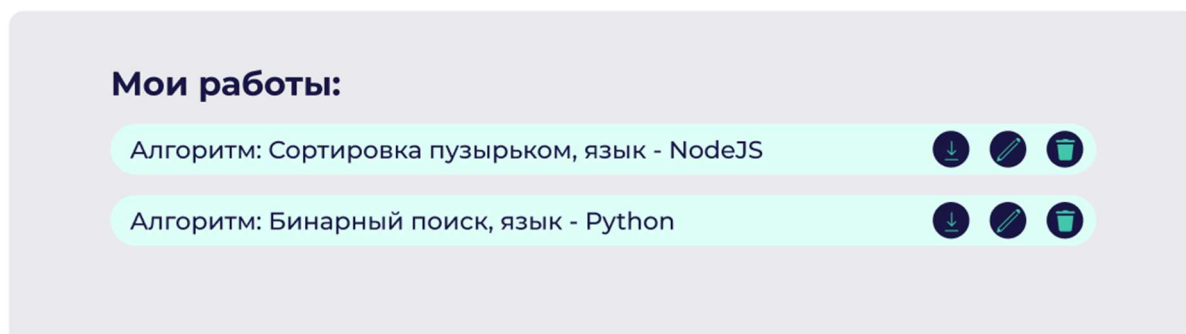
## Компилятор



## Описание алгоритма

### Рисунок 26 – Проверка работы компилятора при ошибке

Для выгрузки отчета о проделанной работе необходимо перейти в личный кабинет, где представлены алгоритмы, с которыми работал пользователь и на каком языке (рис. 27). Также присутствует возможность продолжить над ним работу. Пример, сгенерированного отчета представлен в приложении (см. Приложение А).



### Рисунок 27 – Личный кабинет

Для преподавателя личный кабинет отличается, у него есть возможность создать группу и редактировать ее состав (рис.28-30).

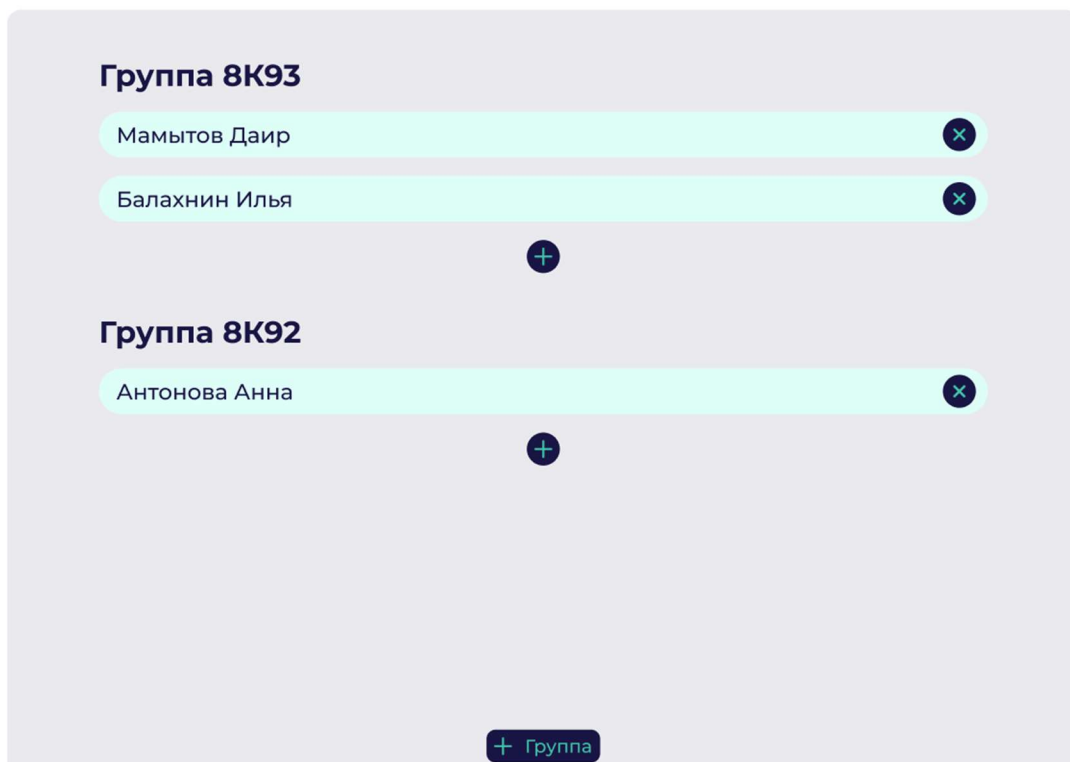


Рисунок 28 – Личный кабинет преподавателя

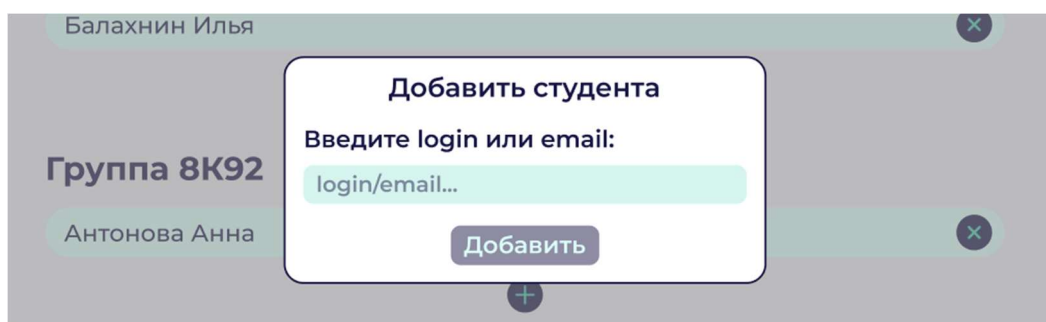


Рисунок 29 – Добавление студента

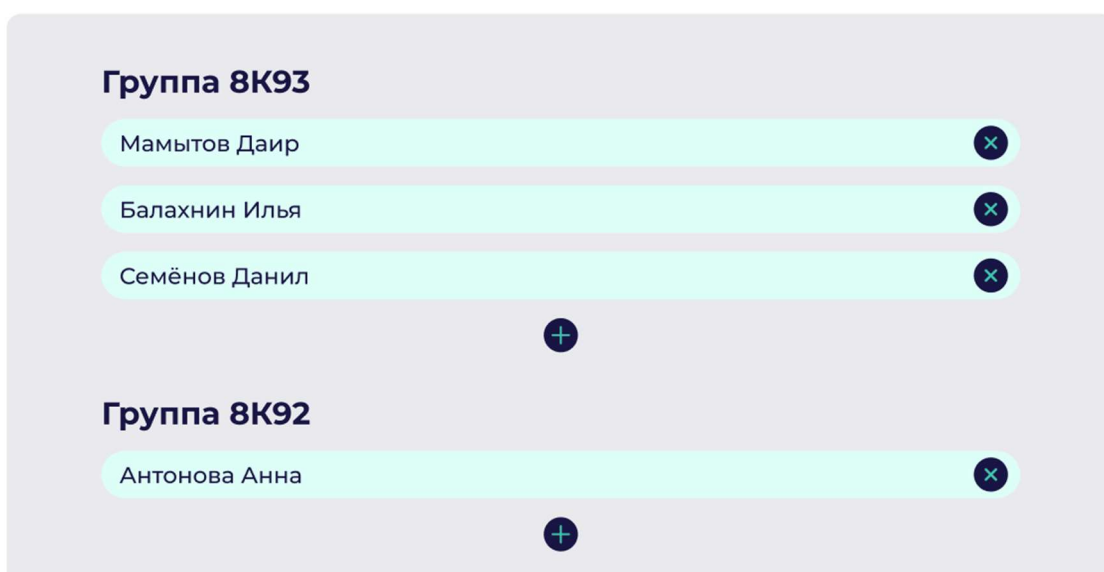


Рисунок 30 – Результат добавления

## ЗАДАНИЕ ДЛЯ РАЗДЕЛА «ФИНАНСОВЫЙ МЕНЕДЖМЕНТ, РЕСУРСОЭФФЕКТИВНОСТЬ И РЕСУРСОСБЕРЕЖЕНИЕ»

Студенту:

<b>Группа</b>	<b>ФИО</b>
8К93	Мамытов Даир

<b>Школа</b>	<b>ИШИТР</b>	<b>Отделение</b>	Отделение информационных технологий
<b>Уровень образования</b>	Бакалавриат	<b>Направление/специальность</b>	09.03.04 Программная инженерия

### Исходные данные к разделу «Финансовый менеджмент, ресурсоэффективность и ресурсосбережение»:

1. Стоимость ресурсов научного исследования (НИ): материально-технических, энергетических, финансовых, информационных и человеческих	Работа с информацией, представленной в российских и иностранных научных публикациях, аналитических материалах, статических бюллетенях и изданиях, нормативно-правовых документах; анкетирование.
2. Нормы и нормативы расходования ресурсов	
3. Используемая система налогообложения, ставки налогов, отчислений, дисконтирования и кредитования	

### Перечень вопросов, подлежащих исследованию, проектированию и разработке:

1. Оценка коммерческого потенциала, перспективности и альтернатив проведения НИ с позиции ресурсоэффективности и ресурсосбережения	Проведение предпроектного анализа. Определение целевого рынка и проведение его сегментирования. Выполнение SWOT-анализа проекта
2. Определение возможных альтернатив проведения научных исследований	Определение целей и ожиданий, требований проекта. Определение заинтересованных сторон и их ожиданий.
3. Планирование процесса управления НИ: структура и график проведения, бюджет, риски и организация закупок	Составление календарного плана проекта. Определение бюджета НИ
4. Определение ресурсной, финансовой, экономической эффективности	Проведение оценки экономической эффективности разработки учебной онлайн-платформы с визуализацией известных алгоритмов

### Перечень графического материала (с точным указанием обязательных чертежей):

<ol style="list-style-type: none"> <li>1. Оценка конкурентоспособности технических решений</li> <li>2. Матрица SWOT</li> <li>3. График проведения и бюджет НИ</li> <li>4. Расчёт денежного потока</li> <li>5. Оценка ресурсной, финансовой и экономической эффективности НИ</li> </ol>	
--	--

<b>Дата выдачи задания для раздела по линейному графику</b>	
---	--

### Задание выдал консультант:

<b>Должность</b>	<b>ФИО</b>	<b>Ученая степень, звание</b>	<b>Подпись</b>	<b>Дата</b>
Профессор	Гасанов Маргеррам Али оглы	д.э.н.		

### Задание принял к исполнению студент:

<b>Группа</b>	<b>ФИО</b>	<b>Подпись</b>	<b>Дата</b>
8К93	Мамытов Даир		

## **4. ФИНАНСОВЫЙ МЕНЕДЖМЕНТ, РЕСУРСОЭФФЕКТИВНОСТЬ И РЕСУРСОСБЕРЕЖЕНИЕ**

### **4.1. Оценка коммерческого потенциала и перспективности проведения научных исследований с позиции ресурсоэффективности и ресурсосбережений**

#### **4.1.1. Потенциальные потребители результатов исследования**

Суть данной выпускной квалификационной работы заключается в разработке онлайн платформы для изучения дисциплины «Алгоритмы и анализ сложности», которая позволяет студентам изучать работы известных алгоритмов сортировки, поиска элементов, обхода дерева, вычисления контрольной суммы на основе визуализированных примеров, также выполнять простые упражнения в онлайн-среде с формированием отчета о проделанной работе. Данная разработка позволяет без участия преподавателя осваивать материал в доступной форме в условиях невозможности очного присутствия, например, в период самоизоляции и карантина.

Целевым рынком нынешней разработки является сфера образования. За последнее время сфера IT претерпела большие изменения, связанные с коронавирусной инфекцией, открыла новые возможности для рынка. Что касается потенциальных потребителей, ими будут выступать образовательные учреждения, нацеленные на повышение качество образования в условиях дистанционного образования.

Сегментировать рынок услуг возможно по степени размеру организации и предназначению интернет-ресурса. Результат сегментирования представлен на таблице 4.

Таблица 4 – Карта сегментирования рынка услуг по разработке интернет-ресурса.

		Предназначение интернет-ресурса		
		Корпоративный сайт	Сайты для проверки полученных знаний	Образовательные платформы
Размер	Крупная			
	Средняя			
	Мелкая			

– Фирма А
  – Фирма Б
  – Фирма В

#### 4.1.2. Анализ конкурентных технических решений

В ходе анализа предметной области было выявлено, что на рынке конкуренты существуют в форме онлайн-курсов, например, Stepik (K1) и LeetCode (K2). Анализ конкурентных решений был проведен с помощью оценочной карты, которая представлена в таблице 5.

Таблица 5 – Оценочная карта сравнения конкурентных разработок

Критерии оценивания	Вес критерия	Баллы			Конкурентоспособность		
		БФ	БК1	БК2	КФ	КК1	КК2
<b>Технические критерии оценки ресурсоэффективности</b>							
чет успеваемости студента	0,15	5	1	4	0,75	0,15	0,6
функционал	0,15	5	2	5	0,75	0,3	0,75
отребность в ресурсах памяти	0,1	4	3	3	0,4	0,3	0,3
скорость работы	0,2	5	3	5	1	0,6	1
<b>Экономические критерии оценки эффективности</b>							
стоимость внедрения в университетах	0,15	3	5	2	0,45	0,75	0,3
редполагаемый срок актуальности разработки	0,05	5	5	5	0,25	0,25	0,25
цена обслуживания	0,1	4	5	4	0,4	0,5	0,4
финансирование разработки	0,1	5	4	5	0,5	0,4	0,5
<b>Итого</b>	1	36	28	33	4,5	3,25	4,1

Анализ конкурентных решений определяется по формуле 1:

$$K = \sum V_i * B_i \quad (1)$$

где  $K$  – конкурентоспособность научной разработки или конкурента;

$B_i$  – вес показателя (в долях единицы);

$B_i$  – балл  $i$ -го показателя.

Основываясь на рассчитанных значениях, можно предположить, что данная разработка имеет конкурентное преимущество по сравнению с другими решениями.

#### 4.1.3. Технология QuaD

Технология QuaD (QUality ADvisor) представляет собой гибкий инструмент измерения характеристик, описывающих качество новой разработки и ее перспективность на рынке и позволяющие принимать решение целесообразности вложения денежных средств в научно-исследовательский проект.

Результат QuaD-анализа исследования представлен в таблице 6.

Таблица 6 – Оценка критериев в соответствии с технологией QuaD

Критерии оценки	Вес критерия	Баллы	Максимальный балл	Относительное значение	Средневзвешенное значение
<b>Показатели оценки качества разработки</b>					
Потребность в ресурсах памяти	0,1	100	100	1	0,1
Надежность	0,1	80	100	0,8	0,08
Инновационная привлекательность	0,15	100	100	1	0,15
Функциональная мощность	0,2	100	100	1	0,2
Простота эксплуатации	0,1	100	100	1	0,1
<b>Показатели оценки коммерческого потенциала разработки</b>					
Цена	0,15	80	100	0,8	0,12
Послепродажное обслуживание	0,1	80	100	0,8	0,08
Предполагаемый срок эксплуатации	0,1	80	100	0,8	0,08
Итого:	1	-	-	-	0,91

Средневзвешенное значение показателя качества и перспективности научной разработки равен 0,91 (попадает в промежуток 0,8 – 1), то есть такая разработка считается перспективной.



#### **4.1.4. SWOT-анализ**

SWOT-анализ — метод стратегического планирования, заключающийся в выявлении факторов внутренней и внешней среды организации и разделении их на четыре категории:

- Strengths (сильные стороны),
- Weaknesses (слабые стороны),
- Opportunities (возможности),
- Threats (угрозы).

К внутренним факторам (то, на что объект способен повлиять), относятся сильные и слабые стороны, а к внешним факторам (то, что влияет на объект извне и не контролируется объектом) – возможности и угрозы.

В рамках данного анализа выявлены сильные и слабые стороны проекта, а также его возможности и угрозы. Результат SWOT-анализа представлен в таблице 7.

Таблица 7 – SWOT-анализ проекта

		<b>Внутренние факторы</b>	
		<b>Сильные стороны:</b> С1. Новизна разработки С2. Гибкость взаимодействия преподавателей и студентов	<b>Слабые стороны:</b> Сл1. Время на визуализацию алгоритмов и необходимость в кадрах для этого Сл2. Возможные недостатки в функционале системы Сл3. Низкая квалификация разработчиков
<b>Внешние факторы</b>	<b>Возможности:</b> В1. Возможность внедрения в других ВУЗах В2. Расширение функционала	В1С1. Новизна разработки увеличивает вероятность внедрения продукта в других ВУЗах и увеличивает популярность продукта	В1Сл3. При внедрении в другие ВУЗы возможно сотрудничество с более опытными разработчиками, готовыми развивать продукт. В2Сл2. При эксплуатации продукта возможно выявление критерий для расширения функционала.
	<b>Угрозы:</b> У1. Трудность внедрения нового алгоритма У2. Вероятность появления нового подхода в образовании и снижение спроса на разработку У3. Проблемы с финансированием	С2У2. Удобство использование, универсальность и развитие компетентного подхода практически исключают появление нового подхода в ближайшем будущем	У1Сл1. Продолжительность наполнения системы алгоритмами усложняет заполнение разработку, что требует дополнительных затрат на кадры, что может заставить некоторые организации отказаться от покупки продукта. У3Сл2. Из-за отсутствия финансирования может возникнуть недостаток кадров для исправления функционала системы. Решение – привлечение студентов для развития разработки

Из матрицы SWOT-анализа можно сделать вывод, что лучшей стратегией развития проекта является привлечение новых кадров для развития разработки, так как из-за новизны продукта возможны недостатки в функционале разработки. В результате привлечения кадров функционал системы будет дорабатываться, что приведет к увеличению возможностей и популярности разработки.

Самой большой потенциальной угрозой является сложность наполнения визуализированными алгоритмами систему из-за больших временных и кадровых затрат. Одним из возможных решений является привлечение студентов ВУЗов для данного мероприятия и развития разработки. Это также позволит студентам получить практический опыт и повысит их квалификацию.

#### **4.2. Инициация проекта**

В рамках процессов инициации определяются изначальные цели и содержание и фиксируются изначальные финансовые ресурсы. Определяются внутренние и внешние заинтересованные стороны проекта, которые будут взаимодействовать и влиять на общий результат научного проекта.

##### **4.2.1. Цели и результаты проекта**

Перед определением целей необходимо перечислить заинтересованные стороны проекта. Информация по заинтересованным сторонам представлена в таблице 8:

Таблица 8 – Заинтересованные стороны проекта

<b>Заинтересованные стороны проекта</b>	<b>Ожидание заинтересованных сторон</b>
Пользователь	Простота в использовании программного продукта
Разработчик	Удачный выполнение проекта и защита
Научный руководитель, студент	Выполненная выпускная квалификационная работа

Цели и результат проекта представлены в таблице 9:

Таблица 9 – Цели и результаты проекта

<p><b>Цели проекта:</b></p>	<ul style="list-style-type: none"> <li>• Собрать набор алгоритмов и изучить их.</li> <li>• Спроектировать функционал в соответствии с требованиями.</li> <li>• Произвести расчет стоимости разработки.</li> <li>• Создать техническое задание и проектные решения.</li> <li>• Разработать веб-решение для обучения алгоритмам.</li> <li>• Произвести тестирование.</li> </ul>
<p><b>Ожидаемые результаты проекта:</b></p>	<p>Успешная разработка онлайн-платформы и представление результатов.</p>
<p><b>Критерии приемки результата проекта:</b></p>	<p>Успешное тестирование функционала в соответствии с функциональным требованием.</p>
<p><b>Требования к результату проекта:</b></p>	<p><b>Требование:</b></p> <ul style="list-style-type: none"> <li>• Выполненные все пункты функционального требования и требования к пользовательскому интерфейсу.</li> <li>• Разработанный функционал полностью соответствует проектным решениям.</li> </ul>

### 4.3. Планирование научно-исследовательских работ

#### 4.3.1. Структура работ в рамках научного исследования

Важным этапом проведения научно-исследовательских работ является необходимость планирования работ, которое включает в себя определение полного перечня работ, а также их распределение между всеми исполнителями проекта. Исполнителями проекта являются студент и руководитель.

Основная работа по разработке и проектированию системы закрепляется за студентом, в то время как руководитель устанавливает цели, задачи, а также контролирует выполнение проделанной студентом работы.

В таблице 10 представлен перечень работ, а также распределение исполнителей по ним.

Таблица 10 – Перечень работ и распределение исполнителей

<b>Основные этапы</b>	<b>№ работы</b>	<b>Наименование работы</b>	<b>Должность исполнителя</b>
Выбор направления работы	1	Выбор научного руководителя	Студент
	2	Составление и утверждение работ	Студент, Руководитель
	3	Составление календарного план-графика	Студент, Руководитель
	4	Подбор и изучение литературы по теме работы	Студент
	5	Анализ предметной области	Студент
Практическая часть бакалаврской работы	6	Проектирование системы	Студент
	7	Разработка онлайн-платформы	Студент
Оценка результатов	8	Согласование и оценка работы с руководителем	Студент, Руководитель
Оформление отчета	9	Выполнение иных работ (финансовый менеджмент, социальная ответственность)	Студент
	10	Подведение итогов, оформление работы	Студент, Руководитель

### 4.3.2. Определение трудоемкости выполнения работ

В данной работе трудовые затраты составляют основную часть стоимости разработки, поэтому необходимо определить трудоемкость работ для каждого из участников исследования.

Определение трудоёмкости выполнения работ осуществляется на основе экспертной оценки ожидаемой трудоёмкости выполнения каждой работы путём расчёта длительности работ в рабочих и календарных днях каждого этапа раб

Трудоемкость оценивается по формуле 2:

$$t_{ож\ i} = \frac{3t_{min\ i} + 2t_{max\ i}}{5} \quad (2)$$

где  $t_{ож\ i}$  – это ожидаемая трудоемкость  $i$ -ой работы (чел-дни),

$t_{min\ i}$  – это минимально возможная трудоемкость выполнения заданной  $i$ -ой работы (оптимистическая оценка) (чел-дни),

$t_{max\ i}$  – это максимально возможная трудоемкость выполнения заданной  $i$ -ой работы (пессимистическая оценка) (чел-дни).

После оценки ожидаемой трудоемкости производится определение продолжительности каждой работы в рабочих днях по формуле 3:

$$T_{p\ i} = \frac{t_{ож\ i}}{Ч_i} \quad (3)$$

где  $T_{p\ i}$  – это продолжительность одной работы (раб-дни),

$t_{ож\ i}$  – это ожидаемая трудоемкость выполнения одной работы (чел-дни),

$Ч_i$  – это численность исполнителей, выполняющих одновременно одну и ту же работу на этом этапе (чел.).

Рассчитанные трудоёмкости приведены в таблице 8.

### 4.3.3. Определение трудоемкости выполнения работ

Для того чтобы в дальнейшем построить график работ с помощью диаграммы Ганта, необходимо также произвести перевод длительности работ из рабочих дней в календарные по формуле 4

$$T_{ki} = T_{pi} \cdot k_{\text{кал}} \quad (4)$$

где  $T_{ki}$  – это продолжительность выполнения  $i$ -й работы в календарных днях;

$T_{pi}$  – это продолжительность выполнения  $i$ -й работы в рабочих днях;

$k_{\text{кал}}$  – это коэффициент календарности, равный 1,225.

Коэффициент календарности  $k_{\text{кал}}$  рассчитывается по формуле 5

$$k_{\text{кал}} = \frac{T_{\text{кал}}}{T_{\text{кал}} - T_{\text{вых}} - T_{\text{пр}}} \quad (5)$$

где  $k_{\text{кал}}$  – коэффициент календарности;

$T_{\text{кал}}$  – количество календарных дней в году;

$T_{\text{кал}}$  – количество выходных дней в году;

$T_{\text{пр}}$  – количество праздничных дней в году.

С учётом того, что календарных дней в 2023 году 365, а сумма выходных и праздничных дней составляет 67 при шестидневной рабочей неделе, коэффициент календарности равен  $k_{\text{кал}} = 1,225$ .

В таблице 11 приведены временные показатели научного исследования.

Таблица 11 – Временные показатели проведения научного исследования



Название работы	Трудоемкость работ			Исполнители	Длительность работ в раб. днях, $T_{pi}$	Длительность работ в кал. днях, $T_{ki}$
	$t_{min}$	$t_{max}$	$t_{ож\ i}$			
Выбор научного руководителя	3	5	3,8	Студент	3,8	4,7
Составление и утверждение работ	2	5	3,2	Студент	1,6	2,0
				Руководитель	1,6	2,0
Составление календарного план-графика	5	7	5,8	Студент	2,9	3,6
				Руководитель	2,9	3,6
Подбор и изучение литературы по теме работы	8	12	9,6	Студент	9,6	11,8
Анализ предметной области	4	6	4,8	Студент	4,8	5,9
Проектирование системы	10	12	10,8	Студент	10,8	13,2
Разработка онлайн-платформы	20	30	24	Студент	24	29,4
Согласование и оценка работы с руководителем	5	9	6,6	Студент	3,3	4,0
				Руководитель	3,3	4,0
Выполнение иных работ (финансовый менеджмент, социальная ответственность)	15	25	19	Студент	19	23,3
Подведение итогов, оформление работы	6	10	7,6	Студент	3,8	4,7
				Руководитель	3,8	4,7

На основе данных, представленных в таблице 11, построены календарный план-график каждого этапа работ в рамках выполняемого проекта, представленная диаграммой Ганта (табл. 12). Стартовой календарной отметкой выбрано начало текущего учебного семестра – 23.01.2023.



Таблица 12 – Диаграмма Ганта проекта

№ Работы	Продолжительность выполнения работ									
	23.01	28.01	02.02	09.02	21.02	27.02	11.03	10.04	19.04	14.05
	- 27.01	- 01.02	- 08.02	- 20.02	- 26.02	- 10.03	- 09.04	- 18.04	- 13.05	- 23.05
1	■									
2		■ ■								
3			■ ■							
4				■						
5					■					
6						■				
7							■			
8								■ ■		
9									■	
10										■ ■

 – Студент  
 – Научный руководитель

#### 4.3.4. Бюджет научно-технического исследования (НТИ)

В состав бюджета входит стоимость всех расходов, необходимых для выполнения работ по проекту. При формировании бюджета используется группировка затрат по следующим статьям:

- материальные затраты,
- затраты на специальное оборудование,
- основная заработная плата исполнителей,
- дополнительная заработная плата исполнителей,
- отчисления во внебюджетные фонды,
- накладные расходы.

##### 4.3.4.1. Расчет материальных затрат НТИ

Данная статья включает стоимость всех материалов, используемых при разработке проекта:

- стоимость сырья и материалов, приобретаемых на стороне,
- стоимость покупных полуфабрикатов и комплектующих,
- стоимость работ и услуг, которые выполняла сторонняя организация,
- стоимость топлива всех видов, приобретаемого со стороны,
- стоимость энергоресурсов всех видов,
- комиссионные вознаграждения, оплата брокерских и иных посреднических услуг.

Расчет материальных затрат осуществляется по формуле 6:

$$Z_m = (1 + k_T) \cdot \sum_{i=1}^m C_i \cdot N_{рас\ xi} \quad (6)$$

где  $m$  – количество видов материальных ресурсов, потребляемых при выполнении научного исследования;

$N_{рас\ xi}$  – количество материальных ресурсов  $i$ -го вида, планируемых к использованию при выполнении научного исследования (шт., кг, м, м<sup>2</sup> и т.д.);

$C_i$  – цена приобретения единицы  $i$ -го вида потребляемых материальных ресурсов (руб./шт., руб./кг, руб./м, руб./м<sup>2</sup> и т.д.);

$k_T$  – коэффициент, учитывающий транспортно-заготовительные расходы.

Данное исследование не использует материальных затрат, кроме электроэнергии, которая затрачивается на электропотребление компьютера (на двоих исполнителей). Так как студент работает 104 дней, а научный руководитель – 14, суммарная работа над исследованием – 118 дней.

В среднем за один час энергопотребление компьютера составляет 116,85 Вт/час или 0,117 кВт/час, что за рабочий день в 8 часов составляет 0,936 кВт/час (примерно 1кВт).

Стоимость электроэнергии в Томской области – 3,16 рубля/кВт.

$k_T = 0$ , так как транспортировка не требуется.

Таким образом,

$$Z_m = (1 + 0) \cdot 3.16 \cdot 118 = 372,88$$

Общая стоимость материальных затрат составила 372,88 рублей.

#### **4.3.4.2. Расчет затрат на специальное оборудование для научных (экспериментальных) затрат**

Данная статья затрат включает в себя затраты на приобретение специального оборудования. Также в эту статью включаются затраты по доставке и монтажу оборудования, равные 15% от его стоимости.

В ходе работы над проектом использовалось оборудование, имеющееся у исполнителей, соответственно необходим расчет его амортизации.

При создании информационной системы был использован персональный компьютера стоимостью 50000 рублей.

Срок полезного использования для машин офисных согласно ОКОФ: код 330.28.23.23 составляет 3 года (36 месяцев). Планируемое время использования ПК для написания ВКР – 5 месяцев.

Амортизация основных средств рассчитывается по формуле 7:

$$A = A_n \cdot OC_{пер} \quad (7)$$

где  $OC_{пер}$  – первоначальная стоимость основных средств;

$A_n$  – норма амортизации.

Норма амортизации основных средств рассчитывается по формуле 8:

$$A_n = \frac{1}{n} \cdot 100\% \quad (8)$$

где  $n$  – установленный срок в месяцах;

Расчет нормы амортизации:

$$A_n = \frac{1}{36} \cdot 100\% = 2,78\%$$

Расчет ежемесячных амортизационных отчислений:

$$A_m = 0,0278 \cdot 50000 = 1389 \text{ руб.}$$

Итоговая сумма амортизации за 5 месяцев:

$$A = 1389 \cdot 5 = 6945 \text{ руб.}$$

#### 4.3.4.3. Основная заработная плата исполнителей темы

В настоящую статью включается основная заработная плата научных и инженерно-технических работников, рабочих макетных мастерских и опытных производств, непосредственно участвующих в выполнении работ по данной теме. Величина расходов по заработной плате определяется исходя из трудоемкости выполняемых работ и действующей системы оплаты труда. В состав основной заработной платы включается премия, выплачиваемая ежемесячно из фонда заработной платы (размер определяется Положением об оплате труда). Расчет основной заработной платы сводится в таблице 13.

Таблица 13 – Расчет основной заработной платы

№ п/п	Наименование этапов	Исполнители по категориям	Трудоемкость, чел.-дн.	Заработная плата, приходящаяся на один чел.-дн., руб	Всего заработная плата по тарифу (окладам), руб.
1		Студент	104	4500	4500
2		Руководитель	14	30000	30000
Итого:					34500

$$Z_{зп} = Z_{осн} + Z_{доп} \quad (9)$$

где  $Z_{осн}$  – основная заработная плата;

$Z_{доп}$  – дополнительная заработная плата.

Основная заработная плата  $Z_{осн}$  работника рассчитывается по следующей формуле:

$$Z_{осн} = Z_{дн} \cdot T_p \quad (10)$$

где  $T_p$  – продолжительность работ, выполняемых научно-техническим работником, раб.дн.;

$Z_{дн}$  – среднедневная заработная плата работника, руб.

Среднедневную заработную плату можно получить по формуле 11:

$$Z_{дн} = \frac{Z_m \cdot M}{F_d} \quad (1)$$

где  $Z_m$  – месячный должностной оклад исполнителя, руб.;

$M$  – количество месяцев работы равно:

При отпуске в 24 рабочих дня  $M = 11,2$  месяца, 5 – дневная неделя;

При отпуске в 48 рабочих дней  $M = 10,4$  месяца, 6 дневная неделя;

$F_d$  – действительный годовой фонд рабочего времени персонала по разработке.

Баланс рабочего времени представлен в таблице 14.

Таблица 14 – Баланс рабочего времени

Показатели рабочего времени	Руководитель	Магистр
Календарное число дней	365	365
Количество нерабочих дней		
- выходные дни	67	67
- праздничные дни		
Потери рабочего времени		
- отпуск	56	56
- невыходы по болезни		
Действительный годовой фонд рабочего времени	242	242

Месячный должностной оклад работника:

$$Z_m = Z_{тс} \cdot (1 + k_{пр} + k_d) \cdot k_p \quad (12)$$

где  $Z_{тс}$  – заработная плата по тарифной ставке, руб.

$k_{пр}$  – премиальный коэффициент (0,3);

$k_d$  – коэффициент доплат и надбавок (0,2-0,5);

$k_p$  – районный коэффициент (для Томска 1,3);

Результаты расчета основной заработной платы приведены в таблице 15.

Таблица 15 – Результаты расчета основной заработной платы

Исполнители	$Z_{mc}$	$k_{np}$	$k_d$	$k_p$	$Z_m$	$Z_{дн}$	$T_p$	$Z_{осн}$ , руб.
Студент	10000	0,3	0,2	1,3	19500	838	104	87154
Руководитель	30000	0,3	0,2	1,3	58500	2514	14	35196
Итого $Z_{осн}$								122350

После расчетов итоговая сумма затрат на основную заработную плату составила 122 350 руб.

#### 4.3.4.4. Дополнительная заработная плата исполнителей темы

Данная статья расходов учитывает величину предусмотренных Трудовым кодексом РФ доплат за отклонение от нормальных условий труда и выплат, связанных с обеспечением гарантий и компенсаций.

Расчёт дополнительной заработной платы осуществляется по формуле:

$$Z_{доп} = k_{доп} \cdot Z_{осн} \quad (13)$$

где  $Z_{доп}$  – дополнительная заработная плата, руб.;

$k_{доп}$  – коэффициент дополнительной зарплаты (на стадии проектирования принимается равным 0,12-0,15);

$Z_{осн}$  – основная заработная плата, руб.

Расчет затрат на дополнительную заработную плату приведен в таблице 16.

Таблица 16 – Затраты на дополнительную заработную плату

Исполнители	$Z_{осн}$ , руб.	$k_{доп}$	$Z_{доп}$ , руб.
Студент	87154	0,12	10458
Руководитель	35197	0,12	4224
Итого			14682

Итоговая сумма затрат на дополнительную заработную плату составила 14 682 руб.

#### 4.3.4.5. Отчисления во внебюджетные фонды (страховые отчисления)

К отчислениям во внебюджетные фонды относятся отчисления:

- отчисления органам государственного социального страхования (ФСС);
- отчисления в пенсионный фонд (ПФ);
- отчисления медицинского страхования (ФФОМС).

Сумма отчислений во внебюджетные фонды рассчитывается на основе затрат на оплату труда исполнителей и может быть вычислена по формуле 14.

$$Z_{\text{внеб}} = k_{\text{внеб}} \cdot (Z_{\text{осн}} + Z_{\text{доп}}) \quad (14)$$

где  $k_{\text{внеб}}$  – коэффициент отчислений на уплату во внебюджетные фонды (пенсионный фонд, фонд обязательного медицинского страхования и др.);

Размер коэффициента определяется законодательно и в настоящее время согласно Федеральному закону от 24.07.2009 №212-ФЗ установлен в размере 30%.

Расчет затрат на отчисления во внебюджетные фонды приведен в таблице 17.

Таблица 17 – Расчет затрат на отчисления во внебюджетные фонды

Исполнители	$Z_{\text{осн}}$ , руб.	$Z_{\text{доп}}$ , руб.	$k_{\text{внеб}}$	$Z_{\text{внеб}}$ , руб.
Студент	87154	10458	0,3	29284
Руководитель	35197	4224	0,3	11826
Итого				41110

После расчетов итоговая сумма отчислений во внебюджетные фонды составила 41 110 руб.

#### 4.3.4.6. Накладные расходы

Накладные расходы – расходы на организацию, управление и обслуживание процесса производства товара, оказания услуги; носят комплексный характер. Накладные расходы вычисляются по формуле 15:

$$Z_{\text{нкл}} = (\text{сумма затрат}) \cdot k_{\text{нр}} \quad (15)$$

где  $k_{нр}$  – коэффициент накладных расходов (16% от суммы затрат, подсчитанных выше).

Расчет накладных расходов приведен в таблице 18.

Таблица 18 – Расчет накладных расходов

Статьи затрат	Сумма, руб
Материальные затраты	372,88
Затраты на специальное оборудование (амортизация)	6945
Затраты на основную заработную плату	122350
Затраты на дополнительную заработную плату	14682
Затраты на отчисления во внебюджетные фонды	41110
<b>Накладные расходы</b>	<b>29674</b>

Итоговая сумма накладных расходов составила 29 694 рублей.

#### **4.3.4.7. Формирование бюджета затрат научно-технического проекта**

После того, как была подсчитана каждая из статей расходов, можно приступить к формированию общего бюджета затрат проекта.

Итоговый бюджет затрат каждой статьи представлена в таблице 19.

Таблица 19 – Расчет бюджета затрат НТИ

Наименование статьи	Сумма, руб
Материальные затраты	373
Затраты на специальное оборудование (амортизация)	6944
Затраты на основную заработную плату	122350
Затраты на дополнительную заработную плату	14682
Затраты на отчисления во внебюджетные фонды	41110
Накладные расходы	29674
<b>Бюджет затрат НТИ</b>	<b>215133</b>

Таким образом, общий бюджет НТИ составляет 215133 рубля.



#### 4.4. Определение ресурсной (ресурсосберегающей), финансовой, бюджетной, социальной и экономической эффективности исследования

В целях определения эффективности исследования рассчитывается интегральный показатель эффективности научно-исследовательского проекта.

Интегральный показатель финансовой эффективности научного исследования получают в ходе оценки бюджета затрат трех (или более) вариантов исполнения научного исследования. Для этого наибольший интегральный показатель реализации технической задачи принимается за базу расчета (как знаменатель), с которым соотносятся финансовые значения по всем вариантам исполнения.

Интегральный финансовый показатель разработки определяется как:

$$I_{\text{финр}}^{\text{исп } i} = \frac{\Phi_{pi}}{\Phi_{\text{max}}} \quad (16)$$

где  $I_{\text{финр}}^{\text{исп } i}$  – интегральный финансовый показатель разработки;

$\Phi_{pi}$  – стоимость  $i$ -го варианта исполнения;

$\Phi_{\text{max}}$  – максимальная стоимость исполнения научно-исследовательского проекта (в т.ч. аналоги).

Полученная величина интегрального финансового показателя разработки отражает соответствующее численное увеличение бюджета затрат разработки в размах (значение больше единицы), либо соответствующее численное удешевление стоимости разработки в размах (значение меньше единицы, но больше нуля).

Интегральный показатель ресурсоэффективности исполнения объекта исследования можно определить по формуле 17:

$$I_{pi} = \sum a_i \cdot b_i \quad (17)$$

где  $I_{pi}$  – интегральный показатель ресурсоэффективности для  $i$ -го варианта исполнения разработки;

$a_i$  – весовой коэффициент;

$b_i$  – бальная оценка, устанавливается экспертным путем по выбранной шкале оценивания;

$n$  – число параметров сравнения.

Для расчета интегрального показателя ресурсоэффективности были выбраны следующие критерии:

- потребность в ресурсах памяти,
- надежность,
- инновационная привлекательность,
- функциональная мощность,
- простота в эксплуатации.

Расчет интегрального показателя ресурсоэффективности приведен в таблице 20.

Таблица 20 – Сравнительная оценка характеристик вариантов исполнения проекта

Критерии	Весовой коэффициент параметра	Текущий проект	Аналог 1	Аналог 2
1. Потребность в ресурсах памяти	0,2	5	4	5
2. Надежность	0,1	3	4	5
3. Инновационная привлекательность	0,3	5	3	2
4. Функциональная мощность	0,3	4	4	3
5. Простота в эксплуатации	0,1	4	3	5
ИТОГО	1	21	18	20

$$I_{\Phi} = \frac{\Phi_i}{\Phi_{max}} = 0.24$$

$$I_{\Phi a1} = \frac{\Phi_{i a1}}{\Phi_{max}} = 0.22$$

$$I_{\Phi a2} = \frac{\Phi_{i a2}}{\Phi_{max}} = 0.25$$

$$I_p = 5 \cdot 0,2 + 3 \cdot 0,1 + 5 \cdot 0,3 + 4 \cdot 0,3 + 4 \cdot 0,1 = 4,4$$

$$I_{p a1} = 4 \cdot 0,2 + 4 \cdot 0,1 + 3 \cdot 0,3 + 4 \cdot 0,3 + 3 \cdot 0,1 = 3,6$$

$$I_{p a2} = 5 \cdot 0,2 + 5 \cdot 0,1 + 2 \cdot 0,3 + 3 \cdot 0,3 + 5 \cdot 0,1 = 3,5$$

$$I_{\text{финр}} = \frac{I_p}{I_{\text{ф}}} = 18,48$$

$$I_{\text{финр а1}} = \frac{I_{p \text{ а1}}}{I_{\text{ф а1}}} = 16,2$$

$$I_{\text{финр а2}} = \frac{I_{p \text{ а2}}}{I_{\text{ф а2}}} = 14$$

$$\mathcal{E}_{\text{ср}} = \frac{I_{\text{финр}}}{I_{\text{финр а1}}} = \frac{18,48}{16,2} = 1,14$$

Общие результаты расчетов для проекта и аналога представлены в виде таблицы (табл. 21).

Таблица 21 – Сравнительная эффективность разработки

№ п/п	Показатели	Проект	Аналог
1	Интегральный финансовый показатель разработки	0,24	0,22
2	Интегральный показатель ресурсоэффективности разработки	4,4	3,6
3	Интегральный показатель эффективности	18,48	16,2
4	Сравнительная эффективность вариантов исполнения	1,14	0,88

Показатель ресурсоэффективности проекта имеет высокий показатель, то есть данная разработка имеет высокий потенциал.

В ходе работы над данным разделом было выявлено, что данная разработка является перспективной по всем рассчитанным параметрам и может быть внедрена в университетах, использующих современные подходы преподавания.

## ЗАДАНИЕ ДЛЯ РАЗДЕЛА «СОЦИАЛЬНАЯ ОТВЕТСТВЕННОСТЬ»

Студенту:

Группа		ФИО	
8K93		Мамытов Даир	
Школа	Инженерная школа информационных технологий и робототехники	Отделение (НОЦ)	Отделение информационных технологий
Уровень образования	Бакалавриат	Направление/специальность	09.03.04 Программная инженерия

Тема ВКР:

<i>Разработка учебной онлайн-платформы с визуализацией известных алгоритмов</i>	
<b>Исходные данные к разделу «Социальная ответственность»:</b>	
<p><b>Введение</b></p> <ul style="list-style-type: none"> <li>– Характеристика объекта исследования (вещество, материал, прибор, алгоритм, методика) и области его применения.</li> <li>– Описание рабочей зоны (рабочего места) при разработке проектного решения/при эксплуатации</li> </ul>	<p><i>Объект исследования:</i> информационная система;  <i>Область применения:</i> образование;  <i>Рабочая зона:</i> офис;  <i>Размеры помещения:</i> 7*8 м<sup>2</sup>.;  <i>Количество и наименование оборудования рабочей зоны:</i> персональные компьютеры (7 шт.), принтер-сканер;  <i>Рабочие процессы, связанные с объектом исследования, осуществляющиеся в рабочей зоне:</i> разработка IT-решения.</p>
Перечень вопросов, подлежащих исследованию, проектированию и разработке:	
<p><b>1. Правовые и организационные вопросы обеспечения безопасности при разработке проектного решения:</b></p> <ul style="list-style-type: none"> <li>– специальные (характерные при эксплуатации объекта исследования, проектируемой рабочей зоны) правовые нормы трудового законодательства;</li> <li>– организационные мероприятия при компоновке рабочей зоны.</li> </ul>	<ul style="list-style-type: none"> <li>– Трудовой кодекс РФ,</li> <li>– ГОСТ 12.2.032-78 Рабочее место при выполнении работ сидя,</li> <li>– СанПин 1.2.3685-21 «Гигиенические нормативы и требования к обеспечению безопасности и (или) безвредности для человека факторов среды обитания».</li> </ul>
<p><b>2. Производственная безопасность при разработке проектного решения:</b></p> <ul style="list-style-type: none"> <li>– Анализ выявленных вредных и опасных производственных факторов</li> </ul>	<p><b>Вредные производственные факторы:</b>  тсутствие и недостаток искусственной освещенности.  мственное перенапряжение, в том числе вызванное информационной нагрузкой.  ревышение уровня шума и другие его неблагоприятные характеристики.  тклонение показателей микроклимата.</p> <p><b>Опасные производственные факторы:</b>  роизводственные факторы, связанные с электрическим током.</p> <p><b>Требуемые средства коллективной и индивидуальной защиты от выявленных факторов:</b>  изоляция проводов, использование наушников и специальных очков при работе с компьютером.</p>
<p><b>3. Экологическая безопасность при разработке проектного решения</b></p>	<p>Воздействие на селитебную зону не выявлено  Воздействие на литосферу выявлено</p>

	<p>Воздействие на гидросферу из-за неверного способа утилизации рабочей техники, канцелярских и прочих отходов</p> <p>Воздействие на атмосферу из-за неверного способа утилизации рабочей техники, канцелярских и прочих отходов</p>
<b>4. Безопасность в чрезвычайных ситуациях при разработке проектного решения</b>	<p><b>Возможные ЧС:</b>          Природные (землетрясения, лесные пожары и т.д.);          Геофизические (землетрясения и т.д.);          Техногенные (транспортные аварии, пожары, аварии с выбросом химически/радиоактивно опасных веществ и т.д.);          Биолого-социального (пандемия)</p> <p><b>Наиболее типичная ЧС:</b> пожар</p>
<b>Дата выдачи задания для раздела по линейному графику</b>	

**Задание выдал консультант:**

Должность	ФИО	Ученая степень, звание	Подпись	Дата
Старший преподаватель	Мезенцева Ирина Леонидовна			

**Задание принял к исполнению студент:**

Группа	ФИО	Подпись	Дата
8К93	Мамытов Даир		

## 5. СОЦИАЛЬНАЯ ОТВЕТСТВЕННОСТЬ

### 5.1. Введение

В данной выпускной квалификационной работе объектом исследования является информационная система для применения в области образования и характер работы программиста веб-сайта, вредные и опасные факторы, влияющие на его деятельность в процессе разработки, описывается производственная безопасность сотрудника, оказываемое воздействие на окружающую среду и экологию в процессе деятельности компании, решения, способствующие сокращению негативного воздействия, описаны способы защиты программиста в случае возникновения чрезвычайных ситуаций и рекомендации по созданию оптимальных условий труда.

Сотрудник, разрабатывающий систему, может являться любым лицом, работающий с персональным компьютером. Процессом является разработка веб-сайта, во время выполнения работником поставленных на него задач на него оказывают влияние следующие факторы: освещение в офисном помещении и непосредственно на рабочем месте; содержание в воздухе неблагоприятных и вредных для организма веществ; шум, источником которого могут являться множество различных приборов и устройств; электромагнитные излучения от оборудования. Рабочим местом сотрудника является стол и находящийся там персональный компьютер с периферией для него.

Для достижения высокой эффективности труда программиста на рабочем месте необходимо сохранять комфортные условия труда за счет сохранения здоровья человека и повышения его физического состояния. Подобные меры важны не только на установленном предприятии, но и на любом другом независимо от сферы деятельности при схожих факторах работы.

## **5.2. Правовые и организационные вопросы обеспечения безопасности**

### **5.2.1. Правовые нормы**

Оплата труда, нормирование ее, выходных и отпусков, условий работы и другие важные аспекты отношений между сотрудником и работодателем представлены в трудовом кодексе Российской Федерации [25]. В том числе продолжительность рабочего времени в течение недели не должна превышать 40 часов, предоставление возможности сокращения рабочего времени.

Также регламентируется длительность сокращенного рабочего времени для различных категорий граждан:

- младше 16 лет – не более 24 часов в неделю;
- в возрасте от 16 до 18 лет и инвалидов I и II группы – не более 35 часов в неделю;
- работающих в условиях труда, которые относятся к вредным 3 или 4 степени или опасных – не более 36 часов в неделю.

Нормируются и работа в ночное время суток, запрещается допускать беременных женщин, детей до 18 лет, инвалидов и работников с детьми-инвалидами, матерей- и отцов-одиночек с детьми до 5 лет, женщин с детьми до 3 лет, осуществляющих уход за больным членом семьи.

В большинстве своём офисные сотрудники относятся к I группе по электробезопасности, им необходимо проходить проверку на знания по работе с предоставленным оборудованием раз в 12 месяцев, помимо этого учитываются и необходимые регулярные медосмотры, которые позволяют выявить симптомы профессиональных заболеваний у сотрудников.

### **5.2.2. Эргономические требования к правильному расположению и компоновке рабочей зоны**

Рабочее место сотрудника должно быть не менее 4,5 квадратных метра при работе с компьютером с плоским монитором, иначе, если монитор является кинескопического типа, тогда не менее 6 квадратных метров [24].

Немаловажным аспектом является то, что при выполнении творческих работ, требующих значительного умственного напряжения или высокой концентрации внимания от сотрудника, рабочие места рекомендуется изолировать друг от друга перегородками. Расстояние от глаз пользователя до монитора должно составлять 600-700 миллиметров, и ни в коем случае не ближе чем 500 миллиметров, учитывая размер отображаемых знаков и символов.

Важная информация, отображаемая на мониторе, будет находиться в пределах, не превышающих 30 градусов от нормальной линии взгляда. В случае использования более чем одного монитора, сторонняя информация будет находиться в пределах 60 градусов.

Стоит учитывать и рабочий стул (кресло), который должен иметь подъемно-поворотный механизм, также присутствовать регулировка по высоте и углам наклона спинки в своей конструкции. Также регулировка каждого перечисленного параметра должна происходить независимо и быть легко осуществимой.

Касательно иной компьютерной периферии на рабочем месте, чем являются мышь и клавиатура. Они должны находиться в пределах оптимального моторного поля сотрудника, непосредственно перед ним, в пределах зоны легкой досягаемости.

При выборе типа рабочего стула (кресла) следует учитывать рост пользователя, физиологию, характер и продолжительности выполнения задач на рабочем месте.



### 5.3. Производственная безопасность

Сотрудники офисов являются большей частью трудоспособного населения. На первый взгляд может показаться, что офис, как и иные места работы, представляет собой абсолютно безопасное помещение, но это не так. На данном рабочем месте воздействует большое количество различных факторов, включая и негативные. Одними из важных наименований оборудования офиса служит персональный компьютер и информационно-коммуникационная техника, которые могут создавать шум. Более подробно факторы описаны в таблице ниже.

Таблица 22 – Возможные опасные и вредные производственные факторы на рабочем месте программиста

Факторы	Нормативный документ
Отсутствие и недостаток необходимой освещенности	СанПиН 1.2.3685-21 «Гигиенические нормативы и требования к обеспечению безопасности и (или) безвредности для человека факторов среды обитания» [1]
Умственное перенапряжение, в том числе вызванное информационной нагрузкой	
Превышение уровня шума и другие его неблагоприятные характеристики	
Отклонение показателей микроклимата	
Производственные факторы, связанные с электрическим током	

### **5.3.1. Отсутствие или недостаток необходимой освещенности**

Для выполнения необходимых задач на рабочем месте требуется уделить немаловажное внимание освещению. Присутствует множество факторов, причиной которых является неверные решения в отношении падающего света, отражаемых поверхностей и других. Стоит выделить несколько важных пунктов оказывающих влияние на человеческий организм в негативную сторону:

- свет отражаемый или прямой от поверхностей,
- мерцание дешевых или некачественных ламп,
- недостаток освещения у рабочего места.

Все вышеперечисленные аспекты способны привести к снижению остроты зрения, раздражению сетчатки глаза и быстрому утомлению работника.

Согласно СанПиН 1.2.3685-21 "Гигиенические нормативы и требования к обеспечению безопасности и (или) безвредности для человека факторов среды обитания" к искусственному и освещению выставлены требования к офисам, КЕО естественного освещения должно составлять 3% при верхнем или комбинированном и 1% при боковом. КЕО совмещенного 1.8 при верхнем или комбинированном и 0.6 при боковом. Параметры искусственного освещения: коэффициент пульсации не более 15%; освещенность при комбинированном всего 400лк, от общего 200лк; освещенность при общем освещении 300лк.

При верно подобранном освещении и мебели появляется возможность поддерживать высокую производительность труда и сохранение физического состояния сотрудника. Мерами для поддержания данных факторов могут служить установка занавесок или козырьков при большом количестве естественного освещения, подбор матовой мебели и верно подобранных ламп.

### **5.3.2. Умственное перенапряжение, в том числе вызванное информационной нагрузкой**

Создание продукта непосредственно связано с работой за персональным компьютером и является стрессовым. Постоянное воздействие данного фактора без необходимой «разгрузки» способен привести к ухудшению психологического состояния сотрудника, а в следствии и к физическим, здоровью.

Умственное перенапряжение способно привести к нервозности, головным болям, нарушению сна, снижению концентрации и другим последствиям. Впоследствии при отсутствии контроля возможны проблемы с пищеварительной системой, сердцем и сосудами.

Устанавливается 3 категории тяжести и напряженности работы с компьютером для трудовой деятельности, которые определяются: для группы А – по суммарному числу считываемых знаков за рабочую смену не более 60000; для группы Б – по суммарному числу считываемых или вводимых знаков за рабочую смену не более 40000; для группы В – по суммарному времени непосредственной работы с компьютером.

При 8-часовой рабочей смене и работе за компьютером следует устанавливать регламентированные перерывы, продолжительность 20 минут дважды – через 1.5-2 часа с начала смены и через 1.5-2 часа после обеденного перерыва. Или 15 минутный через каждый час работы. Также не стоит допускать, чтобы работник пропускал обеденный перерыв.

### **5.3.3. Превышение уровня шума и другие его неблагоприятные характеристики**

Если рассматривать влияние шума на человеческих организм, тогда стоит классифицировать его по источнику возникновения:

- человеческий – шум от других людей и их деятельности,
- технический – шум от работы офисного оборудования,
- внешний – шум с улицы.

Воздействие на организм сотрудника шумом происходит в широком диапазоне частот, а главную опасность представляет шум более высокого уровня, суммированный более чем из одного источника.

В большинстве случаев шум вызывает у человека раздражительность, ухудшению внимания, снижению слуха, упадку настроения или ухудшению сна, воздействует центральную и вегетативную нервную систему. Реже может вызвать нарушение обмена веществ и работы сердечно-сосудистой системы. В результате регулярного воздействия пагубного фактора возможно развитие неврита слухового нерва или «профессиональной тугоухости».

Согласно СанПиН 1.2.3685-21 уровень шума на рабочих местах сотрудника не должен превышать 80 дБА для умственной работы, требующей сосредоточенности.

Предполагаемыми средствами защиты будут служить в первую очередь установка периферии компьютера, создающие меньше шума, например, исключение механических громких клавиатур, затем установка в системные блоки и информационно-коммуникационную технику вентиляторов с поддержкой низкого количества оборотов. Выше перечисленные факторы являются постоянными источниками шума, вследствие чего требуют большого внимания.

### 5.3.4. Отклонение показателей микроклимата

Источником возникновения фактора по большей мере служит отсутствие регулирования температуры воздуха в помещении. Также помимо температуры стоит учитывать такие факторы как: относительная влажность воздуха, скорость движения воздуха. Значения допустимых и оптимальных перечисленных показателей варьируются в зависимости от периода года: теплый и холодный.

Необходимая работа проводится сидя и не требует физического напряжения, сотрудники заняты умственным трудом. При выходе из норм оптимальных параметров микроклимата на рабочем месте состояние служащего ухудшается, повышается утомляемость, что сказывается непосредственно на их работоспособности в компании.

Допустимые значения диапазонов для категории работ Ia по уровню энергозатрат во время холодного периода года:

- температура воздуха ниже оптимальной 20.0-21.9°C,
- температура воздуха выше оптимальной 24.1-25.0°C,
- относительная влажность 15-75%,
- скорость движения воздуха не более 0.1 м<sup>3</sup>.

Параметры для теплого периода года для аналогичной группы:

- температура воздуха ниже оптимальной 21.0-22.9°C,
- температура воздуха выше оптимальной 25.1-28.0°C,
- относительная влажность 15-75%,
- скорость движения воздуха при температуре ниже оптимальной не более 0.1 м<sup>3</sup>, выше – не более 0.2 м<sup>3</sup>.

На рабочем месте при выполнении задач предполагается поддерживать оптимальные параметры микроклимата с помощью техники как в каждом оборудованном офисе, например, кондиционер и увлажнитель воздуха. Соответственно, меры по устранению негативного влияния микроклиматических факторов на сотрудника не требуются.

### **5.3.5. Производственные факторы, связанные с электрическим током**

Персональный компьютер, сетевой принтер, информационно-коммуникационная техника, лампы и так далее, вся вышеперечисленная техника не имеет возможности функционировать должным образом. Таким образом, она является потенциальной причиной возникновения поражения током. Требуется уделить должное внимание данному аспекту – электробезопасность. Основными пунктами в предполагаемых условиях труда на рабочем месте являются:

- мониторинг за состоянием сетевых проводов, розеток, выключателей, вилок и тому подобное,
- контроль и модернизация технической безопасности,
- обучение и проверка знаний персонала по технике безопасности.

Электрический ток, проходящий через тело человека, способен оказывать различные воздействия на организм от нарушения работы мышц и их сокращения до ожогов, перегревание сосудов крови и изменение физико-химического состава крови. Возможны нарушения работы дыхательной и сердечно-сосудистой системы вплоть до прекращения работы. Не стоит исключать и летальные случаи, вследствие получения электрического удара.

Стоит также учесть вредный фактор относительно электрического и магнитного поля, которые замедляют реакцию, ухудшают память, приводят человека к депрессии разной тяжести, повышению раздражительности и нарушению сна.

В соответствии с СанПиНом 1.2.3685-21 значения напряженности электрического поля не должны превышать 0,5 кВ/м, а индукция 10,0 мкТл для общественных зданий. В качестве борьбы предполагается изолировать токоведущие части от случайных прикосаний, поглощение, уменьшение или устранение зарядов статического электричества экранированием источников.

#### **5.4. Экологическая безопасность**

Сам процесс разработки информационной системы учебной онлайн-платформы с помощью персональных компьютеров не оказывает существенного влияния на окружающую среду и гидросферу. На литосферу и атмосферу негативное воздействие несет неправильная утилизация отходов, возникающих в процессе работы.

В офисах работают кондиционеры, которые в свою очередь выделяют фреон и тепло. Накапливаются и канцелярские отходы в процессе документооборота и проектирования. При списании оборудования и техники появляются отходы, требующие специальной утилизации, в том числе аккумуляторы и батареи требуется сдавать в специализированные места. Сотрудники продолжительное время находятся в офисе, соответственно накапливаются продукты жизнедеятельности: недоеденный обед, мусор в санузле и прочее.

Утилизация макулатуры, отходов канцелярии и других отходов осуществляется по средствам сбора, сортировки и переработки. Стоит сделать выбор в сторону многоразовой посуды для уменьшения количества пластикового мусора. Покупка более энергоэффективного оборудования и с более продолжительным сроком службы. Создать культуру в организации для правильной сортировки отходов для последующей переработки, информируя сотрудников о важности данного дела для природы. Вышеперечисленные действия способны сократить негативное влияние на окружающую среду, возможно более обширное влияние при поддержании сознательности и важности подобного дела сотрудниками в домашних условиях.

## 5.5. Безопасность в чрезвычайных ситуациях

К возможным чрезвычайным ситуациям относятся: природные (землетрясения, лесные пожары и т.д.), геофизические (землетрясения и т.д.), техногенные (транспортные аварии, пожары, аварии с выбросом химически/радиоактивно опасных веществ и т.д.) и биолого-социальные (пандемия). С целью уменьшения возможных жертв следует проводить инструктажи по эвакуации и пожаротушения, следовать рекомендациям МЧС. При возникновении вспышки пандемии, по возможности, перевести персонал компании на удаленный формат работы, проводить встречи с использованием технологий видео-конференц-связи.

Наиболее критичная и вероятная чрезвычайная ситуация, которая может случиться – пожар. Связана она с регулярным использованием персональных компьютеров и других видов техники, без которой не представляется работа для разработки, проектирования и поддержки продукта.

Для соответствия пожарной безопасности важно следовать рекомендациям:

- 1) электрооборудование необходимо быть оборудовано теплоизоляционной защитой и подставкой из негорючих материалов;
- 2) офис должен быть оборудован системой автоматической пожарной сигнализации;
- 3) количество запасных выходов и пути эвакуации должны учитывать площадь и общее количество персонала, план эвакуации требуется изобразить наиболее подробным образом, необходимо обозначить пути до лестничных площадок и/или выходов на улицу;
- 4) любой офис обязательно имеет первичные средства пожаротушения – огнетушители, также система противопожарной защиты.

Также можно выделить несколько основных рекомендаций во избежание пожара:

- для предохранения сети от перегрузок запрещается подключать к сети больше потребителей, чем допустимая нагрузка;



- использовать оборудование только при исправном состоянии и целостности электропроводки;
- обеспечить эвакуационные выходы, пути и возможность беспрепятственного прохода людей по ним.

## 5.6. Вывод

Благодаря проведенному исследованию по разделу «Социальная ответственность» были выявлены вредные и опасные факторы, установленные нормативные акты, которые описывают неблагоприятные факторы, оказывающие влияние на сотрудника, меры по предотвращению их негативного воздействия.

Рассмотрены правовые отношения с сотрудником и эргономические требования к рабочей зоне, в которой непосредственно будет выполняться работа.

Проанализированы причины влияния на окружающую среду, теоретические методы их сокращения, что может благоприятно сказаться на культуре компании и ее развитии в какой-либо мере.

Были описаны возможные чрезвычайные ситуации и выявлена наиболее вероятная, пожар. пути их устранения. Рассмотрены меры при возникновении таковых ситуаций и план действий.

Результаты произведенного исследования запланированы использоваться на практике, фактические значения потенциально возможных факторов будут соответствовать нормативным значениям, что благоприятно скажется на производительности труда, удовлетворенности и здоровье работников компании, повысится эффективность самой работы в целом, появляется возможность скорее представить продукт на рынок.

Согласно ПУЭ помещение относится к 1 категории по электробезопасности, температура не превышает 35°C, а влажность воздуха 75%, присутствует приточно-вытяжная вентиляция. Допускаться к обслуживанию электроустановкам будут специалисты с 3-й группой допуска до 1000 вольт.

В соответствии СанПиН 1.2.3685-21 определена Ia категория тяжести труда, работа выполняется сидя и сопровождается незначительными физическим напряжением.

Офисные помещения относятся, как правило, к категории В1-В4 по пожарной опасности, связано это с тем, что присутствуют компьютеры, периферия к ним, кабели интернета, а также мебель, которая выделяет формальдегиды и другие химические вещества при возгорании.

Согласно постановлению Правительства Российской Федерации от 31 декабря 2020 года N 2398 «Критерии отнесения объектов, оказывающих негативное воздействие на окружающую среду, к объектам I, II, III и IV категорий» (с изменением на 7 октября 2021 года) объект, оказывающий минимальное воздействие на окружающую среду относится к IV категории.

## ЗАКЛЮЧЕНИЕ

В результате выполнения работы была разработана онлайн-платформа с возможностью ее дальнейшего совершенствования и дополнения. Реализованы все дополнительные компоненты, которые возможно использовать в различных целях.

Выполнена задача визуализации алгоритмов, с разбиением на компоненты для добавления новых алгоритмов. Таким образом, визуальное представление работы алгоритма работает корректно. Также разработан компилятор с необходимыми компонентами и представлена его работа, окна входа и регистрации с работой токенов. Создана генерация отчетов Word для их последующей правки и сдачи преподавателю.

Система разработана в виде клиент-серверного веб-приложения, где и клиент, и сервер были реализованы благодаря JavaScript, что позволило работать на одном языке. Была возможность задействовать различные библиотеки и модули, которые ускоряют работу и решение поставленных задач. Правильно подобранный стек технологий упростил работу, например, начальная настройка сервера не заняла много времени, чему поспособствовал фреймворк Express.

При возникновении трудностей ответ было несложно найти, разбираясь в устройстве веб-приложения по документации, обширному сообществу разработчиков и форумов с различными видами вопросов и ответов.

В дальнейшем для собственных целей развития планируется, переработать работу личного кабинета, наполнить онлайн-платформу большим количеством алгоритмов, добавить возможности для генерации отчета, например, добавление изображений.

## СПИСОК ИСПОЛЬЗОВАННЫХ ИСТОЧНИКОВ

- 1) MDN Web Docs. – URL: <https://developer.mozilla.org/ru/docs/Web/JavaScript> (дата обращения: 15.02.23). – Текст: электронный.
- 2) Vite | Next Generation Frontend Tooling. – URL: <https://vitejs.dev/> (дата обращения: 24.02.23). – Текст: электронный.
- 3) Node.js. – URL: <https://nodejs.org/en> (дата обращения: 15.02.23). – Текст: электронный.
- 4) Express - фреймворк веб-приложений Node.js. – URL: <https://expressjs.com/ru/> (дата обращения: 20.02.23). – Текст: электронный.
- 5) PostgreSQL: The world's most advanced open source database. – URL: <https://www.postgresql.org/> (дата обращения: 22.02.23). – Текст: электронный.
- 6) Sequelize. – URL: <https://sequelize.org/> (дата обращения: 28.02.23). – Текст: электронный.
- 7) React – A JavaScript library for building user interfaces. – URL: <https://legacy.reactjs.org/> (дата обращения: 05.03.23). – Текст: электронный.
- 8) React-Bootstrap · React-Bootstrap Documentation. – URL: <https://react-bootstrap.github.io/> (дата обращения: 05.03.23). – Текст: электронный.
- 9) Введение | Axios Docs. – URL: <https://axios-http.com/ru/docs/intro> (дата обращения: 09.03.23). – Текст: электронный.
- 10) React Router: Home v6.10.0. – URL: <https://reactrouter.com/en/main> (дата обращения: 08.03.23). – Текст: электронный.
- 11) README MobX. – URL: <https://mobx.js.org/README.html> (дата обращения: 14.03.23). – Текст: электронный.
- 12) Sass: Syntactically Awesome Style Sheets. – URL: <https://sass-lang.com/> (дата обращения: 06.03.23). – Текст: электронный.
- 13) React-vis. – URL: <https://uber.github.io/react-vis/> (дата обращения: 20.03.23). – Текст: электронный.

- 14) Stack Overflow - Where Developers Learn, Share, & Build. – URL: <https://stackoverflow.com/> (дата обращения: 11.02.23). – Текст: электронный. – Текст: электронный.
- 15) Neigh Purr Moo. – URL: <https://www.npmjs.com/package/officegen> (дата обращения: 10.05.23).
- 16) Файловая система NodeJS. – URL: <https://nodejsdev.ru/api/fs/> (дата обращения: 11.05.23). – Текст: электронный.
- 17) Stepik. – URL: <https://stepik.org/catalog> (дата обращения: 22.04.23). – Текст: электронный.
- 18) LeetCode - The World's Leading Online Programming Platform. – URL: <https://leetcode.com/> (дата обращения: 22.04.23).
- 19) Oracle. – URL: <https://www.oracle.com/cis/> (дата обращения: 22.02.23). – Текст: электронный.
- 20) Microsoft SQL Server. – URL: <https://learn.microsoft.com/ru-ru/sql/sql-server/?view=sql-server-ver16> (дата обращения: 22.02.23). – Текст: электронный.
- 21) Angular. – URL: <https://angular.io/> (дата обращения: 05.03.23). . – Текст: электронный.
- 22) Vue. – URL: <https://ru.vuejs.org/index.html> (дата обращения: 05.03.23). – Текст: электронный.
- 23) СанПиН 1.2.3685-21. Гигиенические нормативы и требования к обеспечению безопасности и (или) безвредности для человека факторов среды обитания: дата введения 2021.03.01. — URL: <https://docs.cntd.ru/document/573500115> (дата обращения 02.03.23). – Текст: электронный.
- 24) ГОСТ 12.2.032-78. Система стандартов безопасности труда (ССБТ). Рабочее место при выполнении работ сидя. Общие эргономические требования: дата введения 1979.01.01. — URL: <https://docs.cntd.ru/document/1200003913> (дата обращения 28.02.23). – Текст: электронный.

25) Трудовой кодекс Российской Федерации от 30.12.2001 N 197-ФЗ (редактирования от 25.02.2022) (с изменениями и дополнениями, вступил в силу с 01.03.2022). — URL: [http://www.consultant.ru/document/cons\\_doc\\_law\\_34683/](http://www.consultant.ru/document/cons_doc_law_34683/) (дата обращения 27.02.23). – Текст: электронный.

26) СП 12.13130.2009. Определение категорий помещений, зданий и наружных установок по взрывопожарной и пожарной опасности: дата введения 2009.05.01. — URL: <https://docs.cntd.ru/document/1200071156/> (дата обращения 02.03.23). – Текст: электронный.

## ПРИЛОЖЕНИЕ А

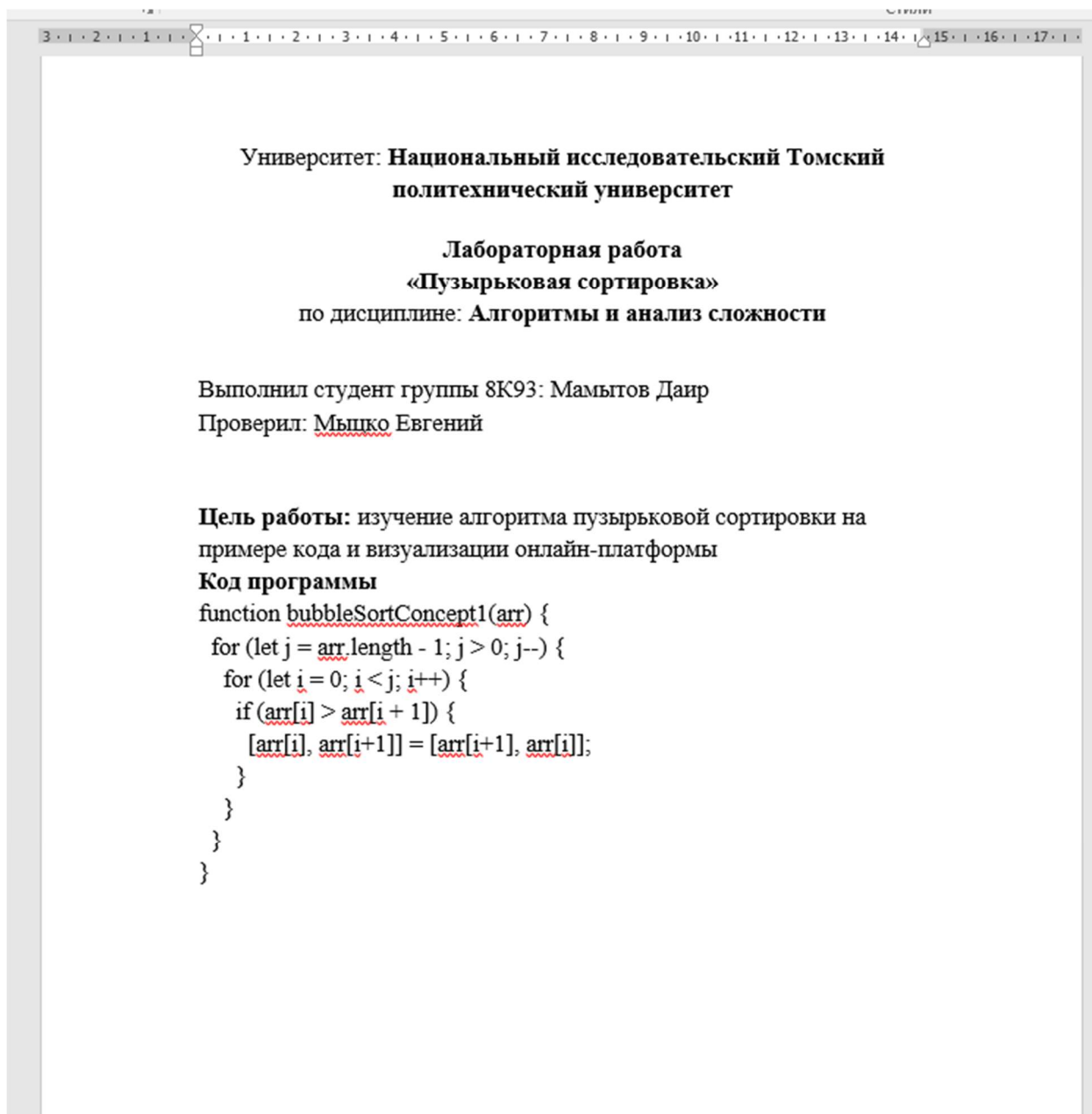


Рисунок А.1 – Пример сгенерированного отчета