



Министерство науки и высшего образования Российской Федерации
федеральное государственное автономное образовательное учреждение высшего образования

«Национальный исследовательский Томский политехнический университет» (ТПУ)
Школа: Инженерная школа информационных технологий и робототехники
Направление подготовки: 09.03.04 «Программная инженерия»
Отделение школы (НОЦ): Отделение информационных технологий

БАКАЛАВРАСКАЯ РАБОТА

Тема работы
Разработка приложения «Конфигуратор ПК»

УДК: 004.94:004.382.7

Студент

Группа	ФИО	Подпись	Дата
8К91	Лепейкин Никита Евгеньевич		

Руководитель

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

КОНСУЛЬТАНТЫ:

По разделу «Финансовый менеджмент, ресурсоэффективность и ресурсосбережение»

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

Томск – 2023 г.

ПЛАНИРУЕМЫЕ РЕЗУЛЬТАТЫ ОСВОЕНИЯ ООП/ОПОП

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

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



Министерство науки и высшего образования Российской Федерации
федеральное государственное автономное образовательное учреждение высшего
образования

Школа: Инженерная школа информационных технологий и робототехники
Направление подготовки (специальность): 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
8К91	Лепейкину Никите Евгеньевич

Тема работы:

Разработка приложения «Конфигуратор ПК»	
<i>Утверждена приказом директора (дата, номер)</i>	№ 102-28 с от 12.04.2023

Срок сдачи обучающимся выполненной работы:	10.06.2023 г.
--	---------------

ТЕХНИЧЕСКОЕ ЗАДАНИЕ:

<p>Исходные данные к работе <i>(наименование объекта исследования или проектирования; производительность или нагрузка; режим работы (непрерывный, периодический, циклический и т. д.); вид сырья или материал изделия; требования к продукту, изделию или процессу; особые требования к функционированию (эксплуатации) объекта или изделия в плане безопасности эксплуатации, влияния на окружающую среду, энергозатратам; экономический анализ и т. д.)</i></p>	<p>Работа направлена на разработку веб-приложения позволяющие создавать «виртуальные» сборки ПК, также проверять готовность и совместимость компонентов в сборке</p>
<p>Перечень разделов пояснительной записки подлежащих исследованию, проектированию и разработке <i>(аналитический обзор литературных источников с целью выяснения достижений мировой науки техники в рассматриваемой области; постановка задачи исследования, проектирования, конструирования; содержание процедуры исследования, проектирования, конструирования; обсуждение результатов выполненной работы; наименование дополнительных разделов, подлежащих разработке; заключение по работе)</i></p>	<ol style="list-style-type: none"> 1. Постановка проблемы и обзор аналогов 2. Проектирование системы конфигуратора ПК 3. Программная реализация веб-приложения конфигуратора ПК 5. Социальная ответственность 6. Финансовый менеджмент, ресурсоэффективность и ресурсосбережение

Перечень графического материала <i>(с точным указанием обязательных чертежей)</i>	Диаграмма вариантов использования, диаграмма деятельности, диаграмма классов, диаграмма последовательности, ER-диаграмма
---	--

Консультанты по разделам выпускной квалификационной работы
(с указанием разделов)

Раздел	Консультант
5. Социальная ответственность	Мезенцева Ирина Леонидовна
6. Финансовый менеджмент, ресурсоэффективность и ресурсосбережение	Гасанов Магеррам Али оглы

Названия разделов, которые должны быть написаны на иностранном языке:

Дата выдачи задания на выполнение выпускной квалификационной работы по линейному графику	
---	--

Задание выдал руководитель / консультант (при наличии):

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

Задание принял к исполнению обучающийся:

Группа	ФИО	Подпись	Дата
8К91	Лепейкин Никита Евгеньевич		



Министерство науки и высшего образования Российской Федерации
федеральное государственное автономное образовательное учреждение высшего
образования

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

КАЛЕНДАРНЫЙ РЕЙТИНГ-ПЛАН
выполнения выпускной квалификационной работы

Обучающийся:

Группа	ФИО
8К91	Лепейкин Никита Евгеньевич

Тема работы:

Разработка приложения «Конфигуратор ПК»

Срок сдачи студентом выполненной работы:	10.06.2023 г.
--	---------------

Дата контроля	Название раздела (модуля) / вид работы (исследования)	Максимальный балл раздела (модуля)
16.06.2023 г.	Изучение проблем самостоятельных сборок	20
16.06.2023 г.	Проектирование системы конфигуратора ПК	25
16.06.2023 г.	Программная реализация веб-приложения конфигуратора ПК	25
16.06.2023 г.	Финансовый менеджмент, ресурсоэффективность и ресурсосбережение	15
16.06.2023 г.	Социальная ответственность	10

СОСТАВИЛ:

Руководитель ВКР

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

СОГЛАСОВАНО:

Руководитель ООП

Должность	ФИО	Ученая степень, звание	Подпись	Дата
Доцент	Чердынцев Евгений Сергеевич	к.т.н.		

Обучающийся

Группа	ФИО	Подпись	Дата
8К91	Лепейкин Никита Евгеньевич		

Реферат

Пояснительная записка содержит 95 страниц (без учета приложений), 32 рисунка, 25 таблиц, 35 источников, 3 приложения.

Ключевые слова: веб-приложение, Laravel, база данных, React, проектирование, PHP.

Объект исследования – веб-приложение для создания «виртуальных» сборок персонального компьютера с возможностью проверять совместимость комплектующих в сборке.

Работа направлена на разработку веб-приложения позволяющим пользователям создавать «виртуальные» сборки ПК, позволяющие:

1. Проверять готовность и совместимость созданной «виртуальной» сборки в режиме онлайн
2. Делиться своими сборками с другими пользователями веб-приложения
3. Создавать отчеты о собранных сборках

Цель работы – разработка веб-приложения с применением фреймворка Laravel.

В ходе работы проводился обзор текущих аналогов, а также непосредственная разработка веб-приложения (проектирование и программная реализация).

Степень внедрения: веб-приложения развернуто на рабочем сервере.

Область применения данного приложения рассчитана на пользователей, планирующих самостоятельно собрать персональный компьютер.

Экономическая эффективность/значимость работы заключается в экономии денежных средств при самостоятельного сборке ПК с использованием данного приложения в качестве справочника.

В результате работы реализовано веб-приложение, позволяющие создавать «виртуальные» сборки ПК.

Основные конструктивные, технологические и технико-эксплуатационные характеристики: языки программирования – PHP (сервер), JS (клиент), фреймворки: Laravel, React, СУБД - MySQL

В будущем планируется улучшить добавить поддержку английского языка, а также расширить список доступных комплектующих.

Содержание

Введение.....	11
Обзор литературы.....	14
1 Постановка проблемы и обзор аналогов.....	15
1.1 Проблемы, с которыми сталкиваются пользователи	15
1.2 Целевая аудитория.....	17
1.3 Обзор аналогов	18
1.3.1 Конфигуратор компьютера «DNS».....	18
1.3.2 Конфигуратор ПК «Citilink».....	19
1.3.3 Конфигуратор ПК «Edelweiss».....	20
1.3.4 Конфигуратор ПК «Регард»	21
1.3.5 Конфигуратор ПК «Xcom shop».....	22
1.4 Сравнение аналогов.....	23
1.5 Выводы по главе	24
1.6 Видение системы	25
1.7 Личный опыт.....	25
2 Проектирование системы конфигуратора ПК.....	26
2.1 Выбор языка программирования и технологий.....	26
2.1.1 Серверная часть	26
2.1.2 База данных	26
2.1.3 Клиентская часть	26
2.1.4 Инструменты для проектирования пользовательского интерфейса.....	27
2.2 Описание основных технических решений	27
2.3 Диаграмма вариантов использования.....	28
2.4 Диаграмма деятельности	35
2.5 Диаграмма последовательности.....	36
2.6 Диаграмма классов	36
2.7 Проектирование базы данных	37
2.7.1 Логическая диаграмма базы данных	39

2.7.2	Физическая диаграмма базы данных	43
2.8	Карта сайта	43
3	Программная реализация веб-приложения конфигулятора ПК	45
3.1	Серверная часть, REST API.....	45
3.2	Сервисный слой и шаблон проектирования «DTO»	46
3.3	Тестирования	47
3.3.1	PHPUnit.....	47
3.3.2	Программная реализация.....	47
3.3.3	Результаты тестирования.....	48
3.4	Разворачивание рабочего сервера.....	49
4	Демонстрация результатов	51
5	Социальная ответственность.....	56
5.1	Правовые и организационные вопросы обеспечения безопасности	56
5.1.1	Правовые нормы трудового законодательства	56
5.2	Производственная безопасность	58
5.2.1	Статические физические перегрузки	59
5.2.2	Умственное перенапряжение и перенапряжение анализаторов, монотонность труда	59
5.2.3	Отсутствие или недостаток необходимого искусственного освещения.....	60
5.2.4	Опасность поражения электрическим током	62
5.3	Экологическая безопасность	62
5.4	Безопасность в чрезвычайных ситуациях	63
5.5	Выводы по главе	64
6	Финансовый менеджмент, ресурсоэффективность и ресурсосбережение	68
6.1	Потенциальные потребители результатов исследования	68
6.2	Анализ конкурентных технических решений.....	68
6.3	Основные и альтернативные способы проведения научных исследований.....	72
6.4	Технологии QuaD	73
6.5	SWOT-анализ	74

6.6	Планирование научно-исследовательских работ	78
6.6.1	Структура работ в рамках научного исследования.....	78
6.6.2	Определение трудоемкости выполнения работ.....	78
6.6.3	Разработка графика проведения научного исследования.....	80
6.7	Бюджет научно-технического исследования	81
6.7.1	Расчет материальных затрат НТИ.....	82
6.7.2	Расчет затрат на специальное оборудование для научных (экспериментальных целей)	82
6.7.3	Основная заработная плата исполнителей темы	83
6.7.4	Дополнительная заработная плата исполнителей темы	86
6.7.5	Отчисления во внебюджетные фонды (страховые отчисления).....	87
6.7.6	Накладные расходы	88
6.7.7	Формирование бюджета затрат научно-исследовательского проекта	88
6.8	Определение потенциального эффекта исследования.....	89
6.9	Выводы по главе	90
	Заключение.....	92
	Список использованных источников.....	93
	Приложение 1.....	96
	Приложение 2.....	97
	Приложение 3.....	103

Введение

Интернет является неотъемлемой частью многих людей. Каждый день с помощью интернета имеется возможность получить необходимую для человека информацию. Но прежде всего нужно иметь рабочую станцию, на которой есть возможность выхода в сеть. На сегодняшний день самой одной из самой популярной станцией являются настольные персональные компьютеры. Есть несколько опций приобрести компьютер.

Первой способ – купить готовый ПК непосредственно в магазине. Данный способ подходит людям, которые имеют низкие технические знания об устройстве компьютером, для этого специалиста заранее собирают ПК и продают уже готовую сборку. Конечному пользователю остается лишь подключить его к источнику питания и всё будет работать. Но у данного способа также есть и недостатки: как правило, за собранные сборки пользователю необходимо переплачивать за услуги сборки. В среднем такая переплата составляет от 20% до 50%. Второй, но не менее важной проблемой является оптимальность сборки. Зачастую поставщики готовых ПК сборок, пытаясь сэкономить на комплектующих, делают достаточно не сбалансированные сборки. Например, нередко в системных блоках с неплохими процессорами обнаруживаются скромные, «офисные» 4 Гб оперативной памяти. Следующая проблема – это гарантия и модульность. Те или иные комплектующие различаются среднестатистической живучестью. Так, например, оперативная память по статистике ломается реже, чем видеокарты, поэтому продолжительная гарантия видеокарт более актуальна. Готовый системный блок придется сдавать на гарантийное обслуживание целиком, что бы в нем ни сломалось. С собственной сборкой все проще: если ломается какая-то деталь, вы просто снимаете ее и несете в сервис. Если без компьютера никак — то вместо ремонтируемой детали ставите запасной аналог и продолжаете работу на время сервисного обслуживания. К тому же свою сборку всегда можно улучшить, тогда как у опломбированного заводского ПК на гарантии такую процедуру можно выполнить только в сервисе [1].

Второй способ – выполнить сборку ПК самостоятельно. Будет необходимо полностью самому собрать свой компьютер. От момента приобретения комплектующих до непосредственной сборки компьютера из комплектующих. У этого способа также есть свои достоинства и недостатки по сравнению с первым способом. Так, например, приобретая комплектующие самостоятельные, можно сэкономить средства, которые могли быть потрачены на готовую сборку, и приобрести более мощные комплектующие. Также перед пользователем открывается широкий выбор ассортимента по комплектующим и имеется возможность собрать оптимальный компьютер. В целом, можно сказать, что все недостатки

рассмотренные в варианте покупки готовый сборки перетекают в достоинства, если рассматривать вариант с готовый сборкой. Но при таком подходе появляется серьезная проблема. Пользователь, который собрать ПК самостоятельно должен иметь уверенные технические знания об устройстве компьютера. Необходимо иметь четкие представление, о том, какие в целом требуется комплектующие для рабочей, какие комплектующие будут совместимы между собой, хватит ли мощности блока питания для исправной работы компьютера. Все эти знания является необходимыми. Любое невыполнение из описанных пунктов придет к тому, что сборка не будет рабочей.

Целью работы является разработка веб-приложения, позволяющая пользователям создавать сборки собственного персонального компьютера, тем самым обеспечить более тщательный подбор комплектующих комплектующие перед покупкой.

Таким образом, это позволит:

1. Упростить процесс сборки ПК рядовым пользователям.
2. Автоматизировать получения данных для отчетов по сборкам ПК для студентов.

Для достижения целей были поставлены следующие задачи:

1. Провести обзор существующих аналогов конфигураторов ПК
2. Показать необходимость разработки собственного конфигуратора
3. Выполнить этап проектирования системы
4. Реализовать конфигуратор ПК в виде веб-приложения, позволяющего создавать сборки ПК в режиме онлайн
5. Выполнить тестирование созданного приложения

Объект исследования – веб-приложение для создания «виртуальных» сборок персонального компьютера с возможностью проверять совместимость комплектующих в сборке.

Метод исследования заключается в анализе литературы и разработке веб-платформы с применением фреймворка Laravel.

Определения, обозначения и сокращения

Работа содержит следующие термины:

Фреймворк: программное средство, для упрощения разработки, содержащее определенный набор инструментов

Регрессионное тестирование: тестирование, направленное на проверку основного функционала программы после очередного релиза.

Клиент: программа, взаимодействующая с сервером.

Сервер: Обслуживающее устройство в системах автоматической обработки информации.

База данных: данные, структурированный в соответствии с определенным порядком и правилами.

Работа содержит следующие сокращения:

ПК – персональный компьютер;

СУБД – система управления базами данных;

REST – representation state transfer;

UI – user interface;

UML – unified modeling language;

Обзор литературы

Для проектирования и реализации веб-приложения были рассмотрены средства разработки и проанализированы фреймворки, языки программирования, системы управления базами данных.

В разработке использованы следующие технологии: JavaScript, React, PHP, Laravel, MySQL.

Особенности языка PHP рассмотрены в источнике [10–11]. Для ознакомления с СУБД MySQL был использован источник [12]. Для разработки клиентской части на языке программирования JavaScript был использован источник [13]. В ходе проектирование системы (включая базу данных) использованы источники [15-18]. Для ознакомления с видами тестирования веб-приложений использованы источники [20-23].

1 Постановка проблемы и обзор аналогов

1.1 Проблемы, с которыми сталкиваются пользователи

1. Отсутствие у рядовых пользователей технических знаний, которые планируют приобрести комплектующие для сборки ПК
2. Большие временные издержки студентов, выполняющих задание в рамках лабораторной работы дисциплины “Основы ЭВМ”, на составление отчетов.

Основной проблемой самостоятельной сборки является низкий уровень знаний пользователей об устройстве работы компьютеров, это затрудняет процесс сборки ПК, т.к. при ручной сборке необходимо учитывать множество критериев, который повлияют на её работоспособность.

Вышесказанную проблему можно решить с помощью т.н. конфигураторов ПК.

Конфигуратор ПК — онлайн-инструмент, который современные поставщики компьютерной техники предлагают использовать своим покупателям. Он позволяет самостоятельно подобрать аппаратные компоненты будущего ПК (процессор, материнскую плату, видеокарту, оперативную память и иные).

Конфигуратор может быть приспособлен для подбора комплектующих системного блока или же компьютера в целом (то есть не только системного блока, но и периферии — монитора, клавиатуры, мыши и прочего). Но чаще всего используется как раз для компоновки системного блока, поскольку его характеристики в первую очередь влияют на общую производительность компьютера. С помощью такого онлайн-инструмента можно подобрать аппаратные компоненты с подходящим для пользователя сочетанием цены и скорости работы [2].

Также стоит отметить, что многие студенты используют конфигураторы ПК при выполнении лабораторной работы, связанной с составлением сборок ПК под различные задачи. Будет удобно, если сервис конфигуратора ПК позволит генерировать отчет о составленной пользователем сборки. Это позволит уменьшить временные издержки при выполнении лабораторной работе и составление отчета по ней.

Для оценки проблемы была составлена диаграмма Исикавы [3].



Рисунок 1 – Диаграммы Исикавы. Создание отчетов по сборке ПК

Поиск комплектующих – студенту требуется искать различные сервисы для поиска комплектующих.

Трудность проверки нестандартизированных отчетов – Так как студенты предоставляют свои отчеты в произвольной форме, преподавателю затруднительно проверять отчеты студентов.

Долгое время на составление отчетов вручную – большие временные издержки, при составлении отчета с нуля.

Отсутствие шаблона – студентам приходится оформлять вид отчета по своему усмотрению.

Большие трудозатраты на учет сопоставимости комплектующих – студенты должны учитывать все детали и нюансы, если они будут собирать сборку полностью самостоятельно. При таком раскладе студент может случайным образом добавить несовместимые комплектующие в сборку или забыть учесть, что мощности блока питания не хватит на стабильную работу сборки.

Проверка состава комплектующих – требуется проверить в наличии необходимый минимум для функционирования сборки

Трудоемкий расчет мощности сборки – для того, чтобы удостовериться о необходимой мощности для работы компьютера требуется знать потребление всех комплектующих.

Из представленной диаграммы можно оценить причины временных издержек при создании отчетов.

1.2 Целевая аудитория

Целевая аудитория онлайн конфигуратора – студенты технических специальностей, а также пользователи, планирующие самостоятельно собрать ПК. В эпоху быстроразвивающихся цифровых технологиях важным фактором в обучении студента является наличие стационарного ПК, который необходим для выполнения различных работ, начиная от лабораторной и заканчивая написанием диссертационной работы.

Одной из главных проблем является тот факт, что у многих студентов возникают различные проблемы при выполнении учебных заданий с помощью программно-аппаратных средств вуза. Это подтверждает статистика из рис.1 [4].

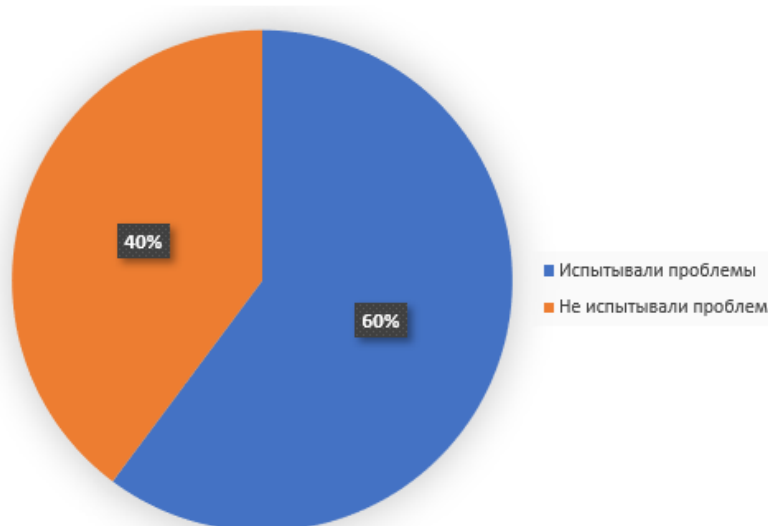


Рисунок 2 – Соотношение респондентов, отличающихся наличием трудностей, которые возникли у них в связи с невозможностью выполнить учебные задания с помощью аппаратно-программных средств вуза

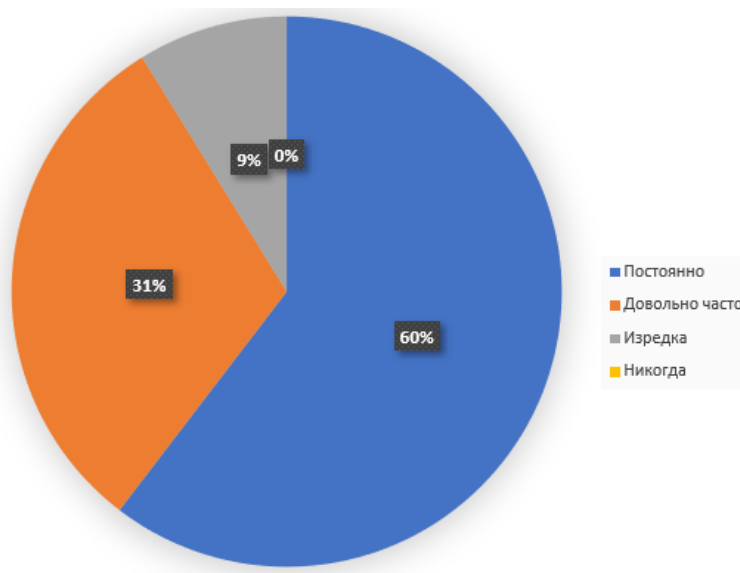


Рисунок 3 – Соотношение студентов, отличающихся периодичностью выполнения учебных заданий на собственных компьютерах (98 респондентов).

Выполнение учебных заданий на собственных компьютерах является неотъемлемой частью процесса обучения у студентов. Об этом говорит статистка, показанная на рисунке 6 [4].

Данные факты подтверждают гипотезу о том, что студентам предпочитают работать на своем собственном компьютере.

Таким образом, основной целевой аудиторией сайта будут студенты, планирующие либо заменить свой старый компьютер на новый, более производительный, либо студенты, которые хотят собрать свой первый компьютер.

1.3 Обзор аналогов

1.3.1 Конфигуратор компьютера «DNS»

Российский интернет-магазин цифровой и бытовой техники.

На сайте интернет-магазина имеется возможность сделать свою собственную сборку, а после ее можно купить в их физическом магазине (при наличии всех комплектующих в ассортименте) [5].

Сайт имеет следующие возможности:

- Регистрация/Авторизация
- Удобный пользовательский интерфейс

- Подробная информация о товаре
- Проверка комплектующих на совместимость
- Возможность сохранить свою сборку и купить ее в их магазине
- Каждый товар имеет свой собственный рейтинг
- Поиск по категориям
- Фильтрация и сортировка
- Имеются пользовательские готовые сборки

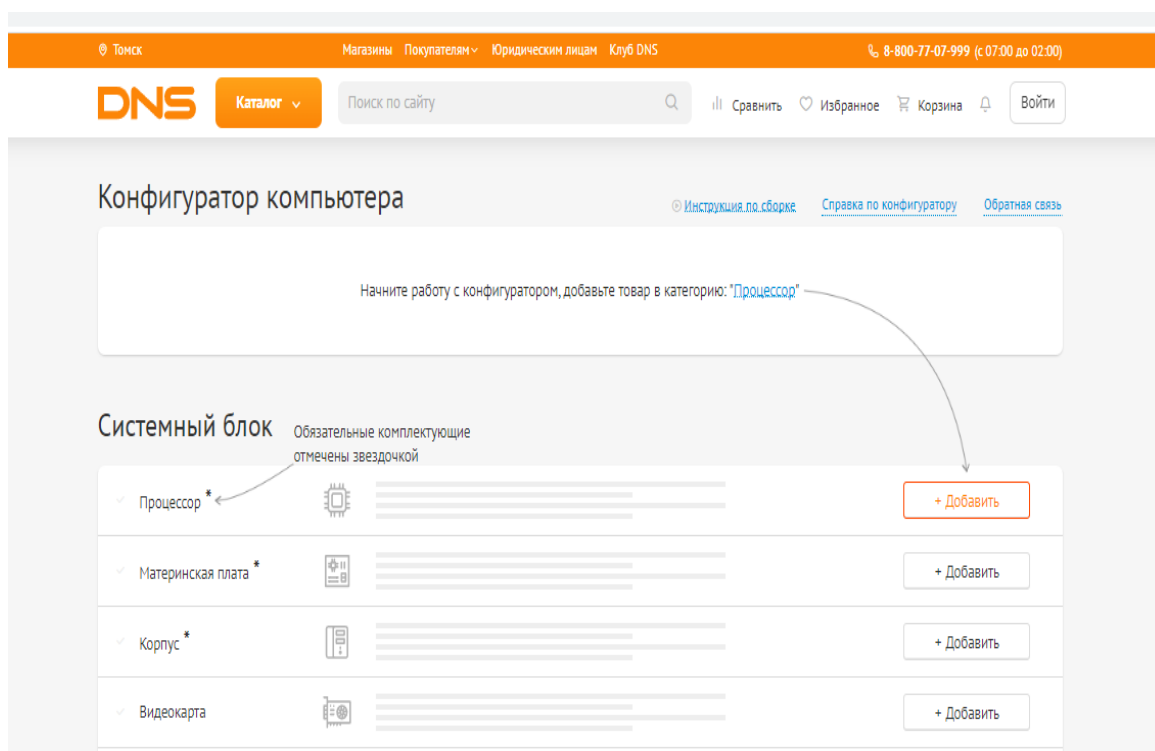


Рисунок 4 – Конфигуратор компьютера DNS

1.3.2 Конфигуратор ПК «Citilink»

Аналогично DNS, Citilink также является российским интернет-магазином цифровой и бытовой техники с возможностью сборки ПК. [6]

Сайт имеет следующие достоинства:

- Регистрация/Авторизация
- Удобный пользовательский интерфейс
- Пользовательские сборки ПК с выбором разделов (игровые, офисные, для работы и тд.)
- Проверка комплектующих на совместимость
- Возможность сохранить свою сборку и купить ее в их магазине
- Каждый товар имеет свой собственный рейтинг

- Поиск по категориям
- Фильтрация и сортировка
- Возможность сохранить свою сборку в формате Excel

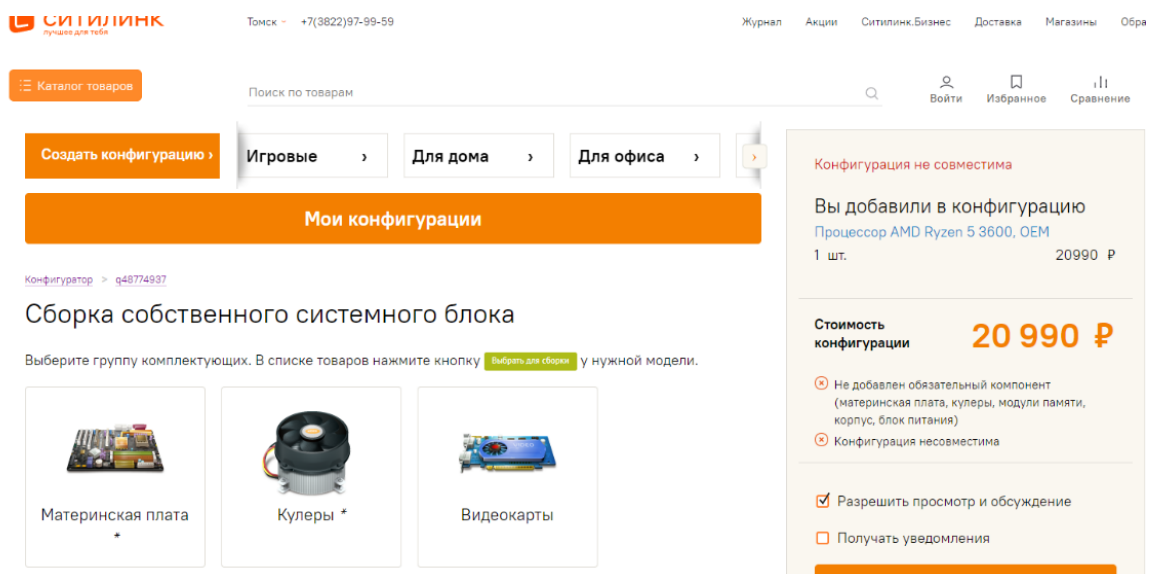


Рисунок 5 – Конфигуратор компьютера Citilink

1.3.3 Конфигуратор ПК «Edelweiss»

EDELWEISS – компьютерная студия, основанная в 2014 году. Главное направление компании производство и тюнинг компьютеров. [7]

Сайт имеет следующие достоинства:

- Регистрация/Авторизация
- Удобный пользовательский интерфейс
- Возможность сохранить свою сборку и купить ее в их магазине
- Проверка комплектующих на совместимость
- Поиск конфигурации по номеру

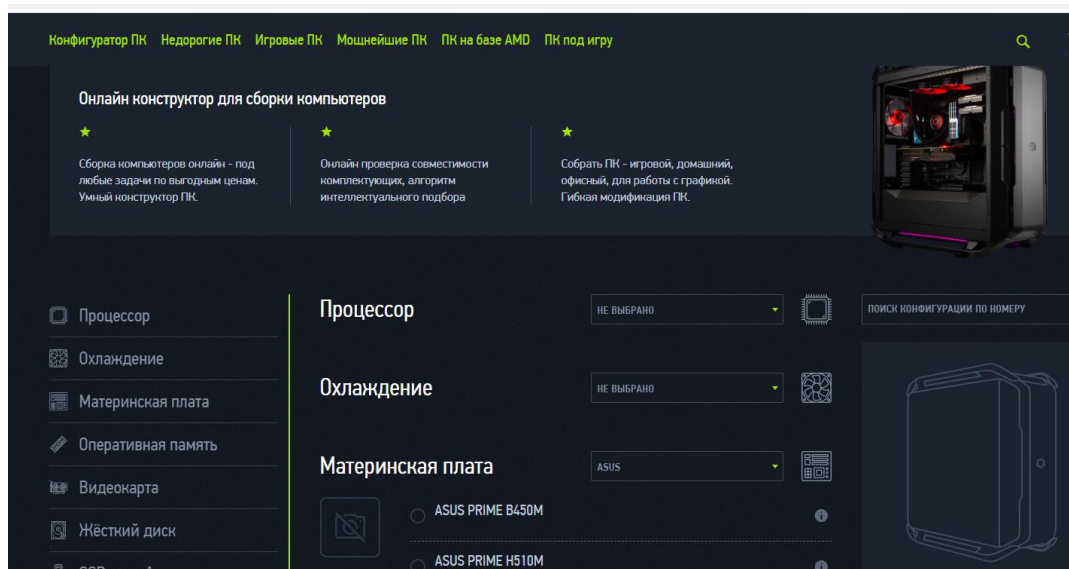


Рисунок 6 – Конфигуратор компьютера Edelweiss

1.3.4 Конфигуратор ПК «Регард»

Регард - интернет-магазин компьютерной, офисной техники и комплектующих. Имеют собственный конфигуратор ПК [8]

Сайт имеет следующие достоинства:

- Регистрация/Авторизация
- Проверка комплектующих на совместимость
- Подробная информация о товаре
- Возможность распечатать собранную конфигурацию
- Возможность сохранить свою сборку и купить ее в их магазине

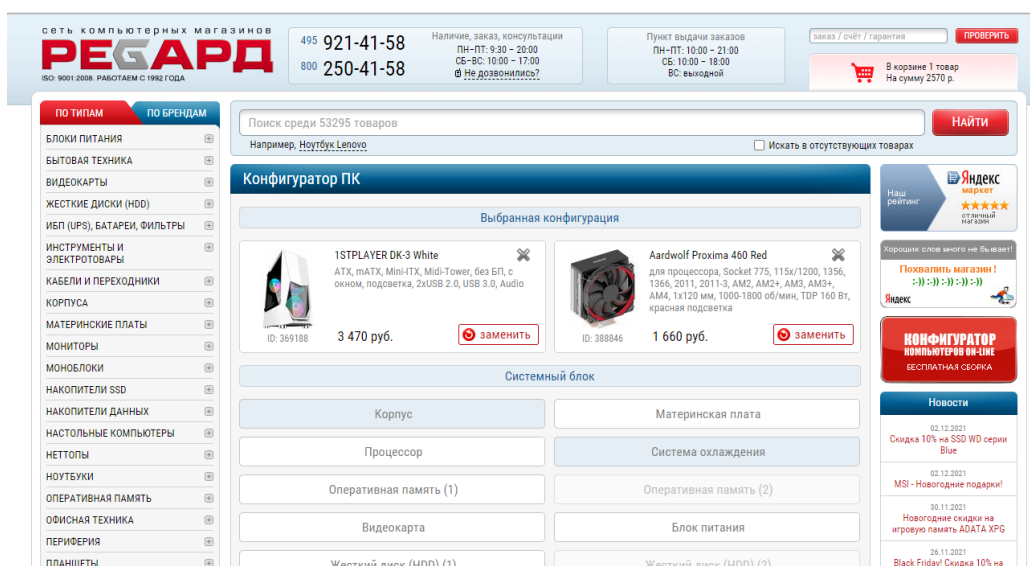


Рисунок 7 – Конфигуратор компьютера Регард

1.3.5 Конфигуратор ПК «Xcom shop»

Xcom shop – является компьютерным интернет-магазином. Имеют собственный конфигуратор ПК [9].

Сайт имеет следующие достоинства:

- Регистрация/Авторизация
- Удобный пользовательский интерфейс
- Проверка комплектующих на совместимость
- Возможность сохранить свою сборку и купить ее в их магазине
- Подробная информация о товаре
- Каждый товар имеет свой собственный рейтинг
- Поиск по категориям
- Фильтрация и сортировка

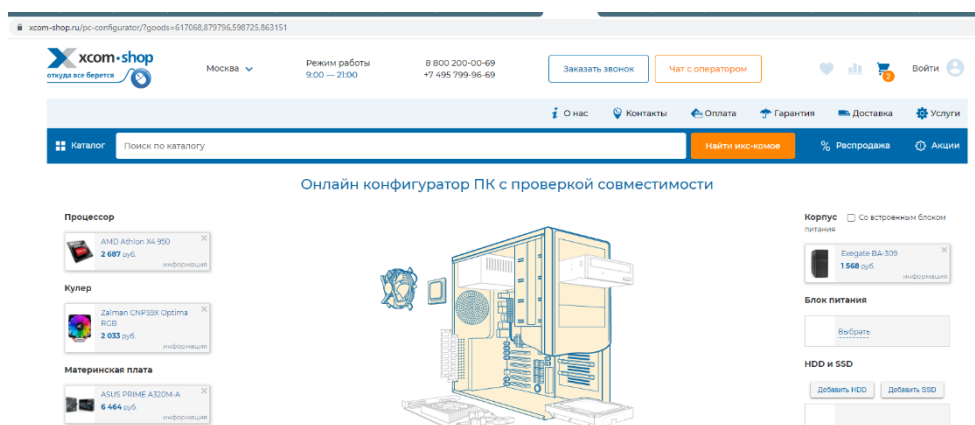


Рисунок 8 – Конфигуратор компьютера Xcom shop

Выводы по обзору

Изучив большинство популярных сайтов с возможностью сделать собственную сборку ПК, можно сделать вывод о том, что многие из них обладают следующими возможностями:

- Регистрация/Авторизация
- Удобный пользовательский интерфейс
- Возможность сохранить свою сборку
- Проверка комплектующих на совместимость
- Пользовательские сборки ПК

- Поиск по категориям
- Фильтрация и сортировка
- Подробная информация о товаре

1.4 Сравнение аналогов

Для сравнения аналогов помимо функциональных возможностей, указанных в разделе «Обзор аналогов», также стоит учитывать возможность **составление отчетов о сборке**.

Рассмотрим следующие критерии сравнения:

- Удобство UI – этот критерий учитывает общее удобство в использовании системы. (диапазон от 1 от 10 баллов).
- Функциональность – связан набором функциональной части сервиса, фильтрация, сортировка и др. (диапазон от 1 от 10 баллов).
- Информация о совместимости – насколько детально системы информирует о проблемах с совместимостью комплектующих (диапазон от 1 от 10 баллов).
- Возможность сохранения сборки – позволяет ли система сохранить созданную сборку (да/нет).
- Создание отчетов – позволяет ли система формировать отчеты о созданных сборках (да/нет)
- Возможность добавления в сборку редких комплектующих (да/нет)

Таблица 1 – Оценка критериев у конкурентов

Аналог\Критерий	Удобство UI	Функциональность поиска	Информация о совместимости	Возможность сохранять сборки	Создания отчетов
DNS	6	9	5	да	нет
Citilink	3	8	1	да	да
Edelweiss	4	2	1	да (на почту)	нет
Регард	7	9	8	да	нет
Xcom shop	4	5	6	нет	нет

Единственный из рассмотренных кандидатов (Конфигуратор компании «Ситилинк») имеет необходимые возможности, которые были указаны в критериях, сохранение сборки, так и формирование отчетов по сборке, но имеет ряд недостатков:

- Некачественная проверка совместимости сборки. В ходе изучения сервиса и создания тестовыхборок, конфигуратор не оповестил пользователя о том, что выбранная сборка является несовместимой.
- Конфигуратор имеет не очень удобный интерфейс, т.к. по неизвестным причинам не позволяет добавить некоторые комплектующие в сборку.

Найденные недостатки не позволяют использовать конфигуратор компании «Ситилинк» в качестве инструмента для решения поставленной задачи.

1.5 Выводы по главе

Все из рассмотренных аналогов предлагали услуги конфигуратора в том числе для дальнейшей покупки собранной сборки во внутреннем магазине системы. Это означает, что ни один из аналогов не имеет возможности добавлять в сборку достаточно редкие

комплектующие, которые, как правило сложно приобрести на рынке (например, процессоры серии «Эльбрус»).

Можно сделать вывод, что ни один из рассмотренных вариантов в полной мере не подходит для решения поставленной задачи, следовательно требуется разработка собственно приложения.

Для дальнейшей работы было составлено техническое задание (приложение 1).

1.6 Видение системы

Система будет представлять собой веб-приложение, помогающая пользователю с выбором комплектующих для своего будущего персонального компьютера. Приложение будет учитывать все нюансы при сборке ПК, а также проверять на совместимость комплектующих между собой. Также пользователи могут составлять сразу несколько сборок (например, в зависимости от бюджета или потребностей в использовании).

Также одной из особенностью приложения будет являться способность формировать отчеты по собранным сборкам. Данная функция поможет студентам автоматизировать выполнение работ, связанных со формированиемборок ПК для различных задач

1.7 Личный опыт

Относительно недавно, в связи с поломкой старого компьютера автора, требовалось приобрести новый, основной задачей было приобрести достаточно мощный компьютер при оптимальной для него цене. С этой задачей помог один из конфигураторов ПК, благодаря нему стало возможным составить оптимальную по мнению автора сборку, как по мощности, так и по цене, а также удостовериться, что собранная сборка является совместимая и пригодной к работе. В следствии чего, оказалось нежным переплачивать за готовую аналогичную сборку.

Данный опыт подчеркивает актуальность сервисов конфигураторов ПК.

2 Проектирование системы конфигуратора ПК

2.1 Выбор языка программирования и технологий

2.1.1 Серверная часть

Для решения поставленной задачи был выбран язык программирования PHP.

PHP (Hypertext PreProcessor, «препроцессор гипертекста») — скриптовый язык программирования. Имеет открытый исходный код. Изначально создавался для разработки веб-приложений, но в процессе обновлений стал языком общего назначения [10].

Данный язык был выбран по причине его простоты в использовании, а также большой популярности среди языков для веб-приложений. PHP занимает седьмое место в этом отчете.

В качестве фреймворка был выбран Laravel также из-за его популярности среди других разработчиков, кроме того, Laravel позволяет достаточно быстро развернуть окружения для непосредственного написания кода, что также является огромным плюсом [11].

2.1.2 База данных

В качестве хранилища данных была выбрана СУБД PostgreSQL. Данный выбор связан с тем, что PostgreSQL является надёжной и простой в использовании СУБД. К тому же она является бесплатной, что делает ее одним из лучших кандидатов.

PostgreSQL – это система управления базами данных (СУБД), распространяемая как свободное программное обеспечение (пользователи имеют право на неограниченную установку, запуск, свободное использование) [12].

2.1.3 Клиентская часть

В качестве языка программирования для работы на клиентской части был выбран JavaScript

JavaScript — мультипарадигменный язык программирования. Поддерживает объектно-ориентированный, императивный и функциональный стили. Является реализацией спецификации ECMAScript (стандарт ECMA-262).

JavaScript обычно используется как встраиваемый язык для программного доступа к объектам приложений. Наиболее широкое применение находит в браузерах как язык сценариев для придания интерактивности веб-страницам [13].

Для программной реализации интерфейса была выбрана библиотека React.

React.js — это библиотека для языка программирования JavaScript с открытым исходным кодом для разработки пользовательских интерфейсов. Она помогает быстро и легко реализовать реактивность — явление, когда в ответ на изменение одного элемента меняется все остальное.

Данный выбор обусловлен высокой гибкостью в использовании, большой популярностью среди разработчиков, а также большого количества пакетов с готовыми UI-компонентами, которые облегчают процесс разработки.

На момент написания работы React входит в тройку популярных библиотек/фреймворков в JavaScript.

2.1.4 Инструменты для проектирования пользовательского интерфейса

В качестве инструмента для прототипирования UI был выбран онлайн-сервис Figma.

Figma — это облачный инструмент для проектирования и прототипирования, который позволяет пользователям создавать и совместно работать над дизайном веб-сайтов, мобильных приложений и других цифровых продуктов [14].

2.2 Описание основных технических решений

В системе используется классическая клиент-серверная архитектура с использованием базы данных. Взаимодействие между клиентом и сервером происходит посредством API запросов от клиента по протоколу HTTP/ HTTPS (рис 9).

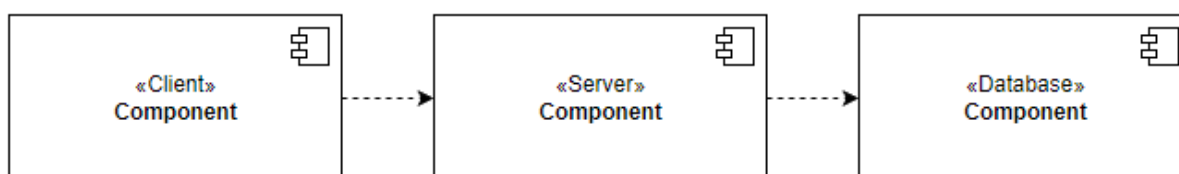


Рисунок 10 – Диаграмма компонентов для системы в целом

Подробнее об компоненте «Server» можно ознакомиться на рисунке 11 .

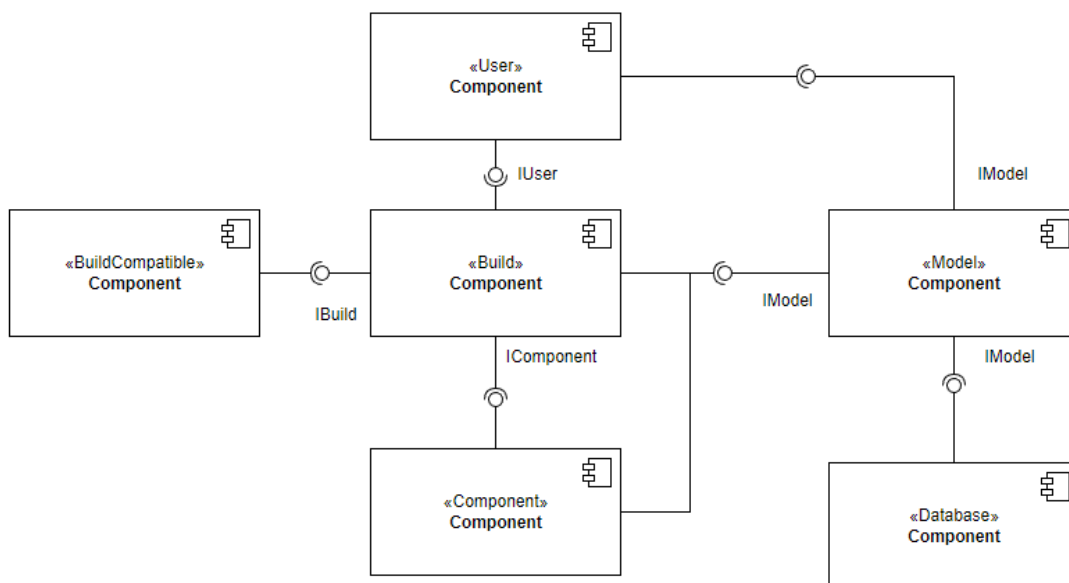


Рисунок 12 – Диаграмма компонентов для серверной части

1. Компонент «User» - модуль ответственный за работу с пользователями системы
2. Компонент «Build» - модуль ответственный за работу со сборками ПК.
3. Компонент «Component» - представляет собой модуль, который отвечает за взаимодействия с комплектующими ПК.
4. Компонент «Model» - модуль, который позволяет создавать сущности для дальнейшей работы с ней.
5. Компонента «Database» - модуль, который взаимодействует с базой данных.
6. Компонент «BuildCompatible» - класс, который проверяет степень готовности сборки ПК.

2.3 Диаграмма вариантов использования

На диаграммах вариантов использования отображается взаимодействие между вариантами использования, представляющими функции системы, и действующими лицами, представляющими людей или системы, получающие или передающие информацию в данную систему. Из диаграмм вариантов использования можно получить довольно много информации о системе. Этот тип диаграмм описывает общую функциональность системы. Пользователи, менеджеры проектов, аналитики, разработчики, специалисты по контролю качества и все, кого интересует система в целом, могут, изучая диаграммы вариантов использования, понять, что система должна делать [15].

На рисунке можно ознакомиться с диаграммой вариантов использования для системы конфигуратора ПК



Рисунок 13 – Диаграмма вариантов использования

Спецификация к диаграмме:

ВИ “Создать сборку”

Цель

Возможность создать ПК сборку, для проверки совместимости компонентов

Актеры

Пользователи,
Конфигуратор

Начальное состояние

Пользователь авторизован
Пользователь перешел в личный профиль

Основной сценарий

1. Пользователь выбрал действие “Создать сборку”

2. Система отобразила интерфейс создания сборки
3. Пользователь начал добавлять комплектующие из списка в сборку
4. Система оповещает пользователя о совместимости или несовместимости сборки.
5. Система подсчитывает стоимость сборки
6. Пользователь выбрал действие сохранить.
7. Система оповестила пользователя об успешном сохранении сборки
8. Система перенаправила пользователя в личный кабинет.

Сценарий обработки ошибки

Предусловие: пользователь не авторизован

1. Система перенаправляет пользователя на страницу Регистрации/Авторизации

ВИ “Удалить сборку”

Цель

Возможность удалить сборку, для того чтобы мои данные по сборкам были актуальными.

Акторы

Пользователь

Начальное состояние

Пользователь авторизован

Пользователь находится в личном профиле

Создана одна сборка или более

Основной сценарий

1. Пользователь выбирает сборку.
2. Пользователь нажимает кнопку “удалить” у сборки.
3. Система запрашивает подтверждения у пользователя об удалении
4. Система удаляет сборку, закрывая окно подтверждения

Альтернативный сценарий

Предусловие: на шаге 3 пользователь отклонил подтверждение об удалении.

5. Система закрывает окно подтверждения, не удаляя сборку

ВИ “Изменить сборку”

Цель

Возможность изменить сборку, для того чтобы данные по сборкам были актуальными.

Акторы

Пользователи

Начальное состояние

Пользователь авторизован

Пользователь находится в личном профиле

Создана одна сборка или более

Основной сценарий

1. Пользователь выбирает сборку
2. Пользователь нажимает кнопку “изменить” у сборки.
3. Система перенаправляет пользователя в окно изменения сборки.
4. Пользователь делает изменения в сборке.
5. Система проверяет совместимости выбранных компонентов
6. Система пересчитывает стоимость сборки
7. Пользователь нажимает кнопку “Сохранить”.
8. Система сохраняет измененную сборку
9. Система оповещает пользователя об успешном сохранении
10. Система перенаправляет пользователя в личный кабинет.

ВИ “Изменить данные пользователя”

Цель

Возможность изменить личные данные пользователя, для того чтобы данные были актуальными.

Акторы

Пользователи

Начальное состояние

Пользователь авторизован

Основной сценарий

1. Пользователь переходит в личный профиль
2. Система отображает интерфейс личного профиля

3. Пользователь изменяет свои личные данные
4. Пользователь нажимает кнопку “Сохранить”
5. Система оповещает пользователя об успешном изменении данных

Сценарий обработки ошибки

Предусловие: на шаге 2 пользователь ввел не валидные данные.

4. Пользователю отображается сообщение, что он ввел неверные данные.
5. Переход на шаг 2 основного сценария с сохранением введенных данных

ВИ “Посмотреть сборки других пользователей”

Цель

Возможность ознакомиться с чужими сборками.

Акторы

Пользователи

Начальное состояние

Система запущена

Основной сценарий

1. Пользователь переходит во вкладку “Посмотреть другие сборки”
2. Система отобразила сборки других пользователей
3. Если пользователю понравилась сборка, он может добавить сборку к себе, выбрав сборку и действие “Добавить сборку”.
4. Система оповестила пользователя об успешном добавлении сборки в профиль.

ВИ “Перейти в личный профиль”

Цель

Возможность посмотреть созданные и добавленные сборки, изменить конфиденциальную информацию

Акторы

Пользователи

Начальное состояние

Пользователь авторизован

Основной сценарий

1. Пользователь переходит в личный профиль при помощи навигационного меню системы.

Сценарий обработки ошибки

Предусловие: пользователь не авторизован

1. Система перенаправляет пользователя на страницу Регистрации/Авторизации

ВИ “Посмотреть все компоненты”

Цель

Возможность посмотреть все компоненты в системе, для того чтобы иметь актуальную информацию о базе компонентов.

Акторы

Менеджер

Начальное состояние

Менеджер авторизован

Основной сценарий

1. Менеджер переходит в раздел компоненты
2. Система отображает список всех компонентов в системе

ВИ “Создать компонент”

Цель

Возможность создать новый компонент в системе, для того чтобы база компонентов была актуальной

Акторы

Менеджер

Начальное состояние

Менеджер авторизован

Основной сценарий

1. Менеджер переходит в раздел “Компоненты”
2. Система отображает список всех компонентов.
3. Менеджер выбирает действие “Добавить новый компонент”
4. Система отображает форму для заполнения нового компонента

5. Менеджер выбирает действие “Сохранить”
6. Система оповещает менеджера об успешном добавлении нового компонента

ВИ “Удалить компонент”

Цель

Возможность удалять компоненты для поддержания базы компонентов в актуальном состоянии

Акторы

Менеджер

Начальное состояние

Менеджер авторизован

В базе компонентов существует хотя бы 1 компонент

Основной сценарий

1. Менеджер переходит в раздел “Компоненты”
2. Система отображает список всех компонентов.
3. Менеджер выбирает компонент
4. Система отображает детальную информацию об компоненте
5. Менеджер выбирает действие “Удалить компонент”.
6. Система оповещает менеджера об успешном удалении компонента.

ВИ “Изменить компонент”

Цель

Возможность изменять компоненты для поддержания базы компонентов в актуальном состоянии

Акторы

Менеджер

Начальное состояние

Менеджер авторизован

В базе компонентов существует хотя бы 1 компонент

Основной сценарий

1. Менеджер переходит в раздел “Компоненты”
2. Система отображает список всех компонентов.
3. Менеджер выбирает компонент
4. Система отображает детальную информацию об компоненте
5. Менеджер изменяет нужны поля в компоненте
6. Система оповещает менеджера об успешном изменении компонента.

2.4 Диаграмма деятельности

Диаграмма деятельности (англ. activity diagram) – диаграмма, на которой показано разложение некоторой деятельности на её составные части. Под деятельностью (англ. activity) понимается спецификация исполняемого поведения в виде координированного последовательного и параллельного выполнения подчинённых 2 элементов – вложенных видов деятельности и отдельных действий (англ. action), соединённых между собой потоками, которые идут от выходов одного узла к входам другого.

Диаграммы деятельности используются при моделировании бизнес - процессов, технологических процессов, последовательных и параллельных вычислений. Аналогом диаграмм деятельности являются схемы алгоритмов по ГОСТ 19.701 - 90 и дракон – схемы [16].

Ниже на рисунке представлена диаграмма деятельности для процесса «Создание сборки ПК»

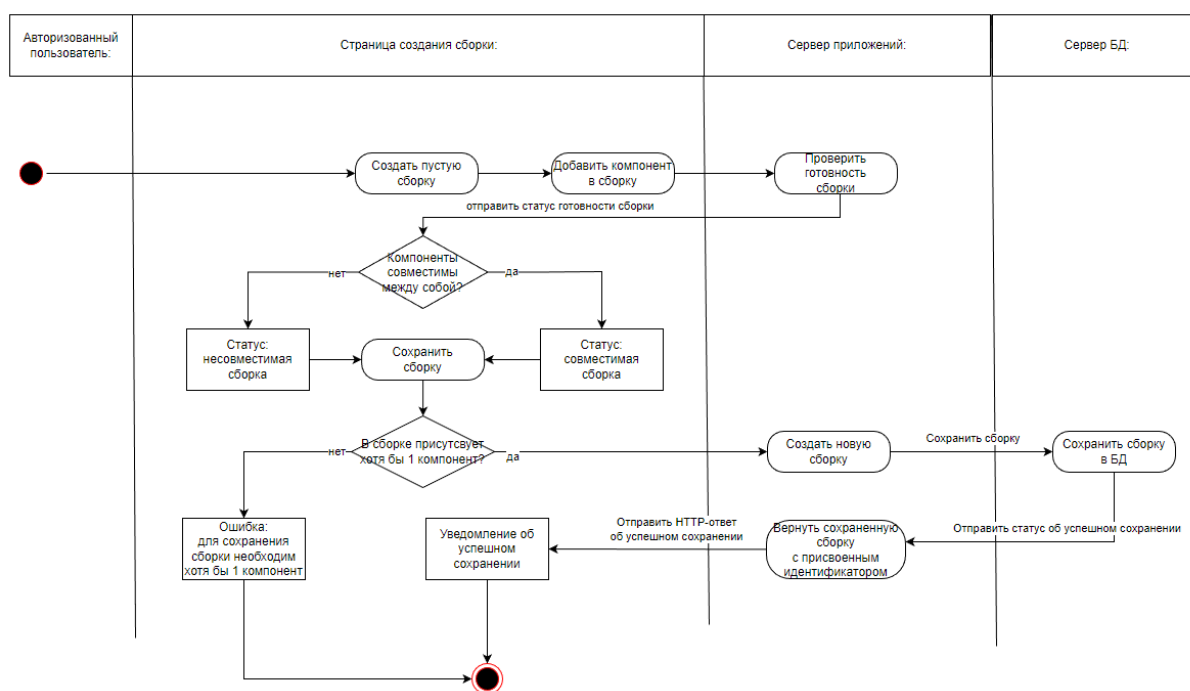


Рисунок 14 – Диаграмма деятельности для процесса «Создание сборки ПК»

Как можно увидеть из диаграммы, всякий раз, когда пользователь добавляет комплектующие к себе в сборку клиентская часть обращается к серверу с целью уточнения, является ли текущая сборка (или, иначе говоря, набор комплектующих) совместимым.

Как видно из диаграммы, прежде чем отправить сборку на сохранения, клиентская часть проверяет наличие в сборке хотя бы одного компонента, это необходимо для фильтрации сборок-пустышек, т.е. в которых отсутствуют какие-либо компоненты и, по сути, не имеют никакого смысла.

2.5 Диаграмма последовательности

Диаграммы последовательности предназначена для моделирования взаимодействия объектов программной системы во времени (жизненный цикл), а также обмена сообщениями между ними в рамках какого - либо определённого прецедента.

Ниже на рисунке представлена диаграмма последовательностей для прецедента «Просмотр пользовательских сборок»

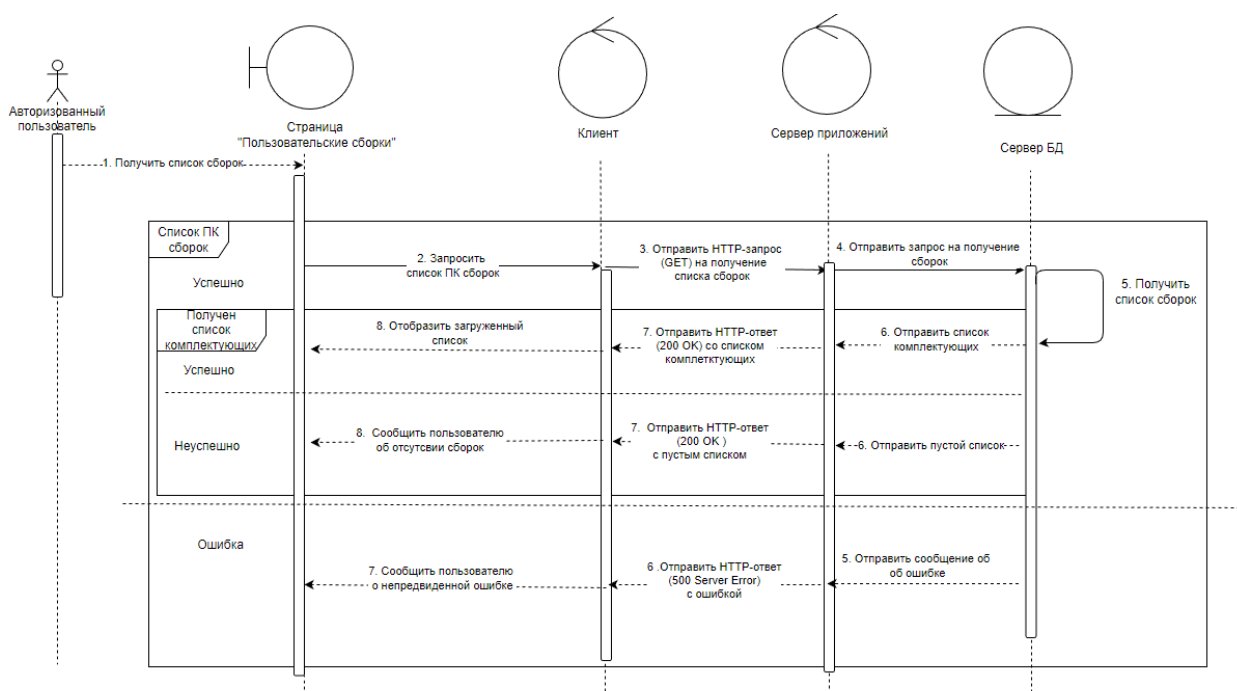


Рисунок 15 – Диаграмма последовательностей

2.6 Диаграмма классов

Диаграмма классов определяет типы классов системы и различного рода статические связи, которые существуют между ними. На диаграммах классов изображаются также атрибуты классов, операции классов и ограничения, которые накладываются на связи между классами. Вид и интерпретация диаграммы классов существенно зависит от точки зрения

(уровня абстракции): классы могут представлять сущности предметной области (в процессе анализа) или элементы программной системы (в процессах проектирования и реализации) [17].

На рисунке 36 (прил. 1) представлена диаграмма классов для системы конфигуратора ПК

На диаграмме представлен набор классов, которые наследуются от абстрактного класса Model, благодаря наследованию от абстракции, подклассы получают базовые операции такие как: просмотр, удаление, обновление, создания и другие.

Далее на диаграмме идет так называемый «сервисный слой» (service layer). Здесь находится вся бизнес-логика работы системы. Именно через классы-сервисы клиент имеет возможность взаимодействовать с моделями, а не напрямую.

Данный подход имеет следующие преимущества:

Во-первых, благодаря таким классам, сами модели не засоряется разными методами, которые, по сути, не имеют отношения к этому классу. Также при таком подходе, будет четкое понимание, где содержатся вся логика работы с моделями, это делает систему более прозрачной, что является несомненным плюсом.

Подробнее об этом можно почитать в главе «Сервисный слой и шаблон проектирования «DTO»

После сервисного слоя, идут контроллеры, данные классы принимают входящий запрос (Request) и возвращают ответ (Response). Как правило в контроллерах проходит валидация данных и преобразования в классы DTO.

2.7 Проектирование базы данных

Проектирование базы данных — процесс создания схемы базы данных и определения необходимых ограничений целостности.

Основные задачи:

- Обеспечение хранения в БД всей необходимой информации.
- Обеспечение возможности получения данных по всем необходимым запросам.
- Сокращение избыточности и дублирования данных.
- Обеспечение целостности базы данных.

Логическое проектирование — создание схемы базы данных на основе конкретной модели данных, например, реляционной модели данных. Для реляционной модели данных логическая модель — набор схем отношений, обычно с указанием первичных ключей, а также «связей» между отношениями, представляющих собой внешние ключи.

Физическое проектирование — создание схемы базы данных для конкретной СУБД. Специфика конкретной СУБД может включать в себя ограничения на именование объектов базы данных, ограничения на поддерживаемые типы данных и т. п. Кроме того, специфика конкретной СУБД при физическом проектировании включает выбор решений, связанных с физической средой хранения данных (выбор методов управления дисковой памятью, разделение БД по файлам и устройствам, методов доступа к данным), создание индексов и т. д [18].

2.7.1 Логическая диаграмма базы данных

На рисунке можно ознакомиться с разработанной логической диаграммой базы данных.

я

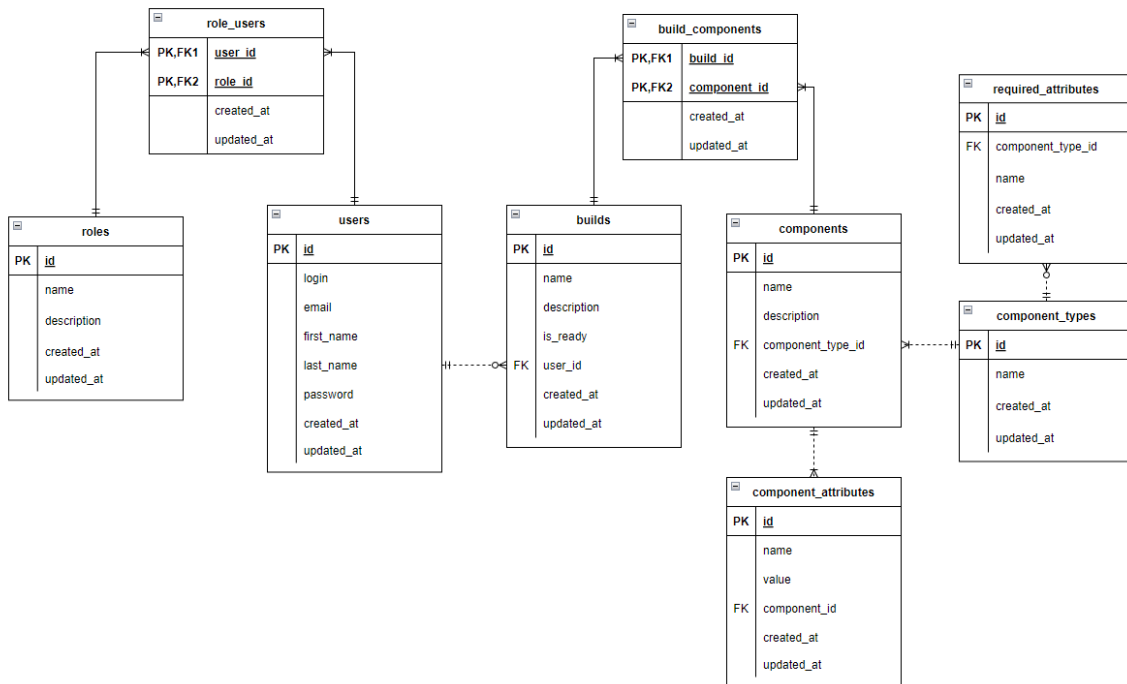


Рисунок 16 – Логическая диаграмм БД

Пояснение к диаграмме

Таблица 2 – Пояснение к логической диаграмме

Столбец	Значение	Тип данных	Ограничения и комментарий
Таблица «users» - пользователи			
user_id	Идентификатор пользователя	INT – числовое значение	PRIMARY KEY – первичный ключ NOT NULL - обязательное поле AUTO INCREMENT - счетчик
email	Электронная почта	VARCHAR(100) – текстовое поле	UNIQUE – уникальное значение
first_name	Имя	VARCHAR(100) – текстовое поле	NOT NULL - обязательное поле
last_name	Фамилия	VARCHAR(100) – текстовое поле	NOT NULL - обязательное поле
password	Пароль	VARCHAR(100) – текстовое поле	NOT NULL - обязательное поле

Столбец	Значение	Тип данных	Ограничения и комментарий
created_at	Дата создания	TIMESTAMP – от-метка времени	
updated_at	Дата обновления	TIMESTAMP – от-метка времени	
Таблица «roles» - роли			
role_id	Идентификатор роли	INT – числовое значение	PRIMARY KEY – первичный ключ NOT NULL - обязательное поле AUTO INCREMENT - счетчик
description	Описание роли	VARCHAR(200) – текстовое поле	
created_at	Дата создания	TIMESTAMP – от-метка времени	
updated_at	Дата обновления	TIMESTAMP – от-метка времени	
Таблица «role_user» - роли у пользователя			
role_id	Идентификатор роли	INT – числовое значение	PRIMARY KEY – первичный ключ FOREIGN KEY – внешний ключ NOT NULL - обязательное поле
user_id	Идентификатор пользователя	VARCHAR(100) – текстовое поле	PRIMARY KEY – первичный ключ FOREIGN KEY – внешний ключ NOT NULL - обязательное поле
Таблица «builds» - сборки			
build_id	Идентификатор сборки	INT – числовое значение	PRIMARY KEY – первичный ключ NOT NULL - обязательное поле AUTO INCREMENT - счетчик
name	Название сборки	VARCHAR(100) – текстовое поле	NOT NULL - обязательное поле
description	Описание сборки	TEXT – текстовое поле	

Столбец	Значение	Тип данных	Ограничения и комментариев
is_ready	Готовность сборки	TINYINT(1) – числовое значение	0 – сборка не готова 1 – сборка готова NOT NULL - обязательное поле
user_id	Идентификатор владельца сборки	INT – числовое значение	NOT NULL - обязательное поле FOREIGN KEY – внешний ключ
created_at	Дата создания	TIMESTAMP – метка времени	
updated_at	Дата обновления	TIMESTAMP – метка времени	
Таблица «components» - компоненты			
component_id	Идентификатор компонента	INT – числовое значение	PRIMARY KEY – первичный ключ NOT NULL - обязательное поле AUTO INCREMENT - счетчик
name	Название компонента	VARCHAR(100) – текстовое поле	NOT NULL - обязательное поле
description	Описание компонента	TEXT – текстовое поле	
type_id	Идентификатор типа компонента	INT – числовое значение	NOT NULL - обязательное поле FOREIGN KEY – внешний ключ
created_at	Дата создания	TIMESTAMP – метка времени	
updated_at	Дата обновления	TIMESTAMP – метка времени	
Таблица «build components» - компоненты сборки			
build_id	Идентификатор сборки	INT – числовое значение	PRIMARY KEY – первичный ключ FOREIGN KEY – внешний ключ NOT NULL - обязательное поле
component_id	Идентификатор компонента	INT – числовое значение	PRIMARY KEY – первичный ключ FOREIGN KEY – внешний ключ NOT NULL - обязательное поле
Таблица «component attributes» - атрибуты компонента			
attribute_id	Идентификатор атрибута	INT – числовое значение	PRIMARY KEY – первичный ключ

Столбец	Значение	Тип данных	Ограничения и комментарий
			NOT NULL - обязательное поле AUTO INCREMENT - счетчик
name	Название атрибута	VARCHAR(100) – текстовое поле	NOT NULL - обязательное поле
value	Значение атрибута	VARCHAR(100) – текстовое поле	NOT NULL - обязательное поле
component_id	Идентификатор компонента	INT – числовое значение	NOT NULL - обязательное поле FOREIGN KEY – внешний ключ
created_at	Дата создания	TIMESTAMP – отметка времени	
updated_at	Дата обновления	TIMESTAMP – отметка времени	
Таблица «component types» - тип компонента			
component_type_id	Идентификатор типа компонента	INT – числовое значение	PRIMARY KEY – первичный ключ NOT NULL - обязательное поле AUTO INCREMENT - счетчик
name	Название типа компонента	VARCHAR(100) – текстовое поле	NOT NULL - обязательное поле
created_at	Дата создания	TIMESTAMP – отметка времени	
updated_at	Дата обновления	TIMESTAMP – отметка времени	
Таблица «required attributes» - обязательные атрибуты для типа компонента			
required_attributes_id	Идентификатор атрибута	INT – числовое значение	PRIMARY KEY – первичный ключ NOT NULL - обязательное поле AUTO INCREMENT - счетчик
name	Название типа компонента	VARCHAR(100) – текстовое поле	NOT NULL - обязательное поле
created_at	Дата создания	TIMESTAMP – отметка времени	
updated_at	Дата обновления	TIMESTAMP – отметка времени	

2.7.2 Физическая диаграмма базы данных

Ниже на рисунке представлена физическая диаграмма базы данных

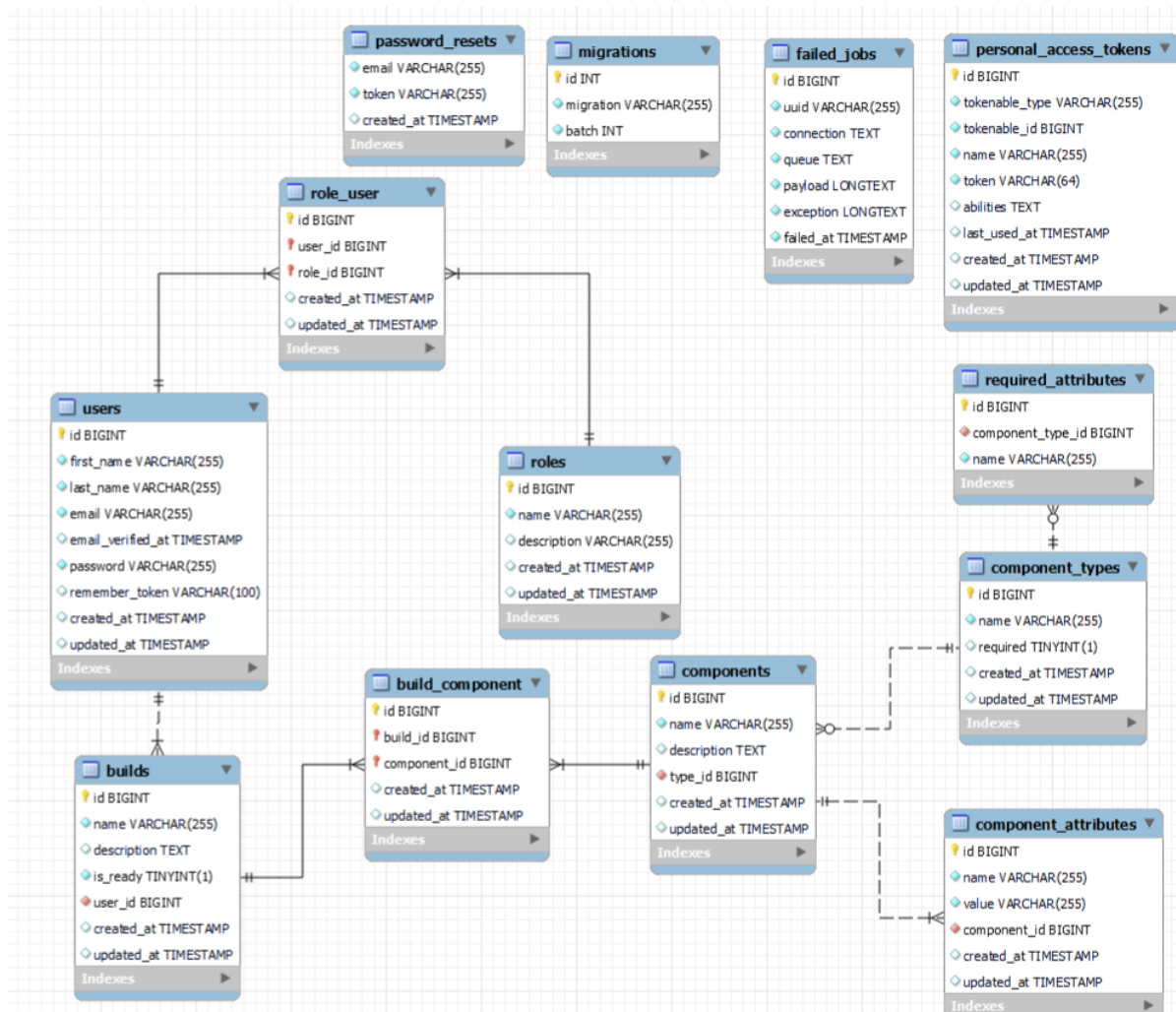


Рисунок 17 – Физическая диаграмма БД

На диаграмме добавилось несколько дополнительных таблиц, которые не были отображены в логической диаграмме. Это служебные таблицы фреймворка Laravel, которые необходимы для корректной работы некоторых функциональных возможностей фреймворка, поэтому указывать их в логической диаграмме не имеет смысла.

2.8 Карта сайта

Карта сайта — наглядная многоуровневая модель структуры веб-сайта, которая помогает пользователям ориентироваться на странице, быстро искать и находить нужную информацию. Обычно карты сайтов служат для упорядочения страниц по определенной теме, а также ее систематизации по уровню иерархии. Чем дальше страничка от основного раздела, тем более специализировано ее содержание.

В рамках данной работы автором была спроектирована карта сайта для конфигураторе ПК (рис.).

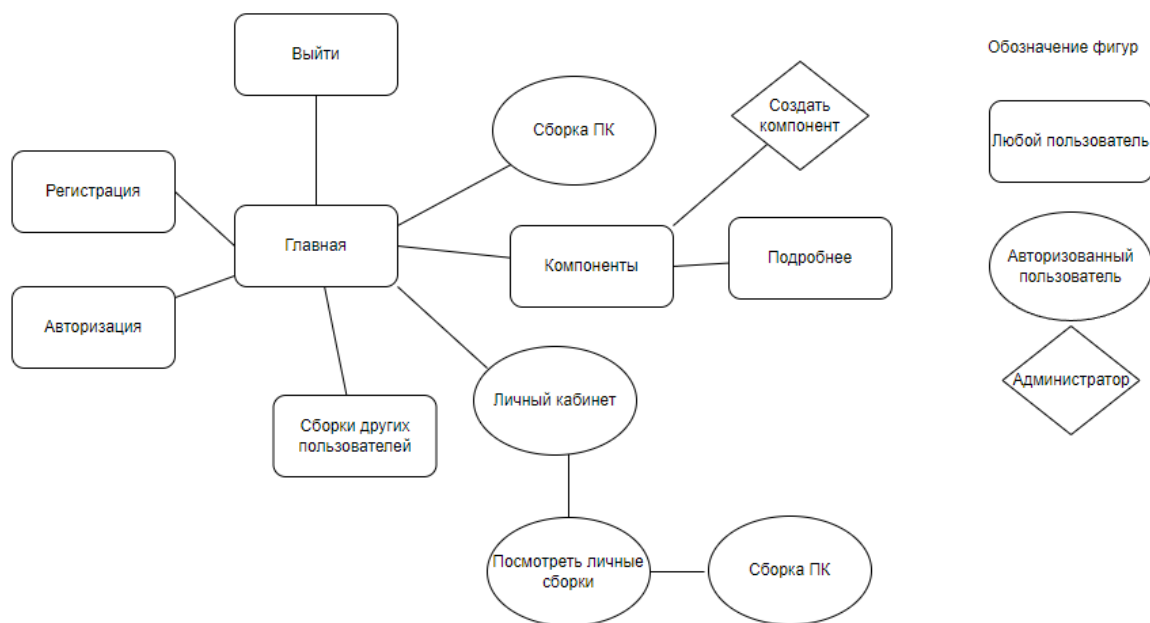


Рисунок 18 – Карта сайта конфигуратора ПК

3 Программная реализация веб-приложения configurator ПК

3.1 Серверная часть, REST API

Как уже было сказано во второй главе приложение использует клиент-серверную архитектуру. Стоит отметить, что сервер написано по принципу REST API

RESTful API — это интерфейс, используемые между клиентом и сервером для безопасного обмена информацией через протокол HTTP/HTTPS. Большинство бизнес-приложений должны взаимодействовать с другими внутренними и сторонними приложениями для выполнения различных задач. Например, чтобы генерировать ежемесячные платежные ведомости, ваша внутренняя бухгалтерская система должна обмениваться данными с банковской системой вашего клиента, чтобы автоматизировать выставление счетов и взаимодействовать с внутренним приложением по учету рабочего времени. RESTful API поддерживают такой обмен информацией, поскольку они следуют безопасным, надежным и эффективным стандартам программного взаимодействия (см рис. ниже) [19].

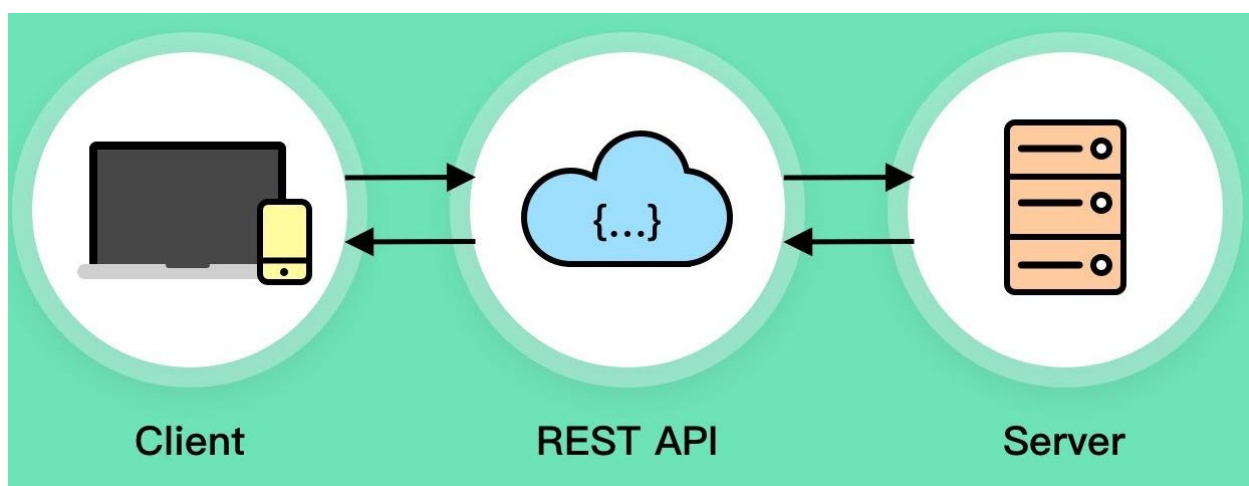


Рисунок 19 – Иллюстрация взаимодействия REST API приложений

Документирование API велось в платформе Postman

Postman – это инструмент для работы с API, который позволяет документировать её удобным способом в виде коллекции. Особенностью программы является ее простота в использовании, а также широкий функционал.

Имеется возможность создавать переменные окружения и менять их в зависимости от программной среды.

На рисунке ниже представлена собранная коллекция API-эндпоинтов configurator ПК

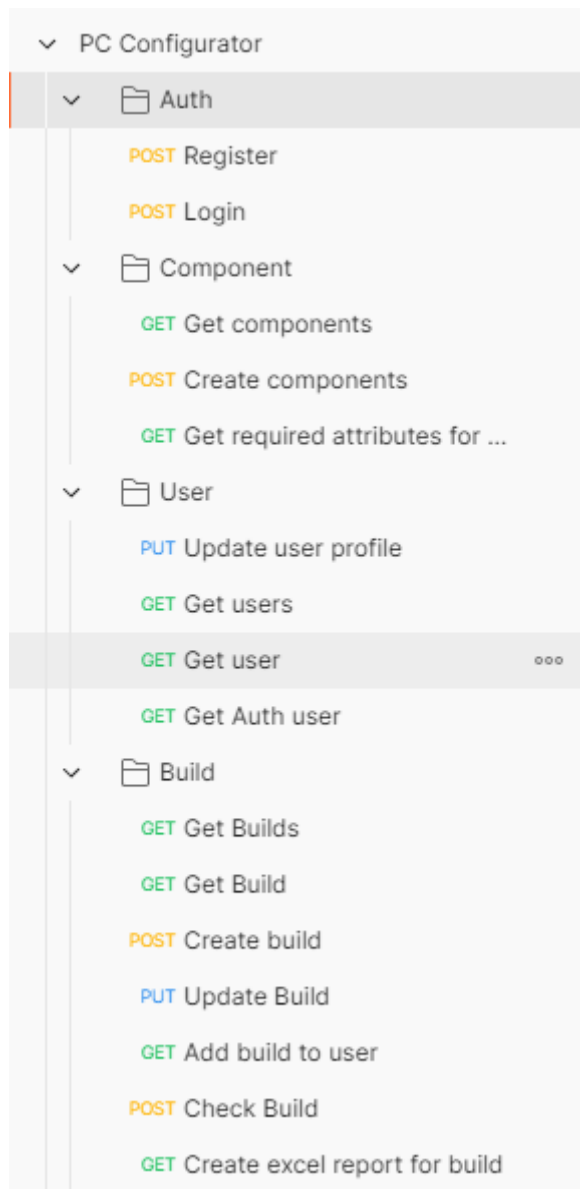


Рисунок 20 – Коллекция эндпоинтов в программе Postman

3.2 Сервисный слой и шаблон проектирования «DTO»

Сервисный слой (Service layer) — это шаблон проектирования, который инкапсулирует бизнес-логику приложения и определяет границу и набор допустимых операций с точки зрения взаимодействующих с ним клиентов.

В разрабатываемом приложении активно используется данный шаблон, как в серверной части, так и в клиентской.

Пример сервисного класса можно увидеть на рисунке ниже.

3.3 Тестирования

Тестирование программного обеспечения — процесс исследования, испытания программного продукта, имеющий своей целью проверку соответствия между реальным поведением программы и её ожидаемым поведением на конечном наборе тестов, выбранных определённым образом [20].

Тестирование конфигулятора ПК производилась методом черного ящика. Тестирование по стратегии чёрного ящика — это процесс тестирования системы и её поведения вне зависимости от её внутренней структуры, архитектуры и реализации [21].

Само тестирования производилось посредством разворачивания тестового сервера, создания тестовой базы данных и отправки API-запросов к серверу по полученным ответам от сервера можно судить о работоспособности данных эндпоинтов.

3.3.1 PHPUnit

Тестирование API исполнялось при помощи PHP-фреймворка для тестирования PHPUnit.

PHPUnit — это система для юнит-тестирования приложений, написанных на языке PHP. Под "юнит" понимаются небольшие блоки кода, например отдельные методы класса. Т.е. можно протестировать метод на работоспособность в автоматическом режиме. Когда приложение достаточно большое, содержащее много классов, методов и тем более если планируется дальнейшее его расширение стоит заняться тестированием и в этом поможет PHPUnit. Он представляет собой отдельную библиотеку классов, которые нужно подключить при создании своего теста [22].

3.3.2 Программная реализация

Laravel — предоставляет удобный интерфейс для создания юнит и интеграционных тестов. Для этого достаточно лишь создать класс для тестов и отнаследоваться от абстрактного класса TestCase [23].

Для более удобного проведение тестов API приложения, был создан еще один абстрактный класс APITest (который также наследовался от TestCase).

Данный класс конфигурирует заголовки запросов, указывает тип принимаемого ответа, а также добавляет токен авторизации.

В результате, для того чтобы создать тест API-запроса необходимо создать класс и отнаследовать его от класса APITest, который в свою очередь подготовит всё необходимое для проведения тестирования API-запроса.

В дочернем классе осталось лишь создать нужное количество публичных методов, начинающихся с ключевого слова test (например, testGetUser). Один метод означает один тест.

С реализацией абстрактного класса можно ознакомиться в приложение 2.

3.3.3 Результаты тестирования

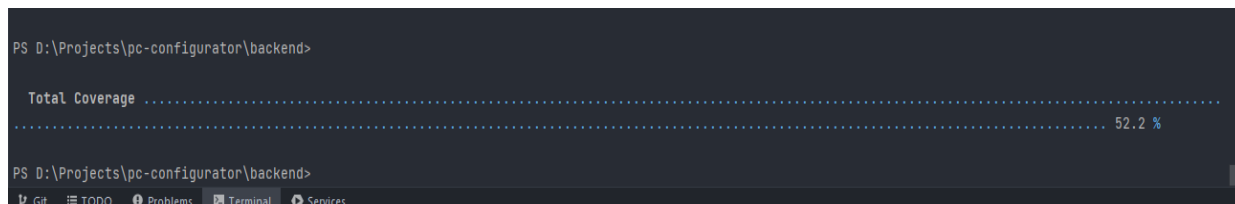
По итогу благодаря проведенному тестированию автору удалось обнаружить несколько ошибок в приложении, которые в следствии чего успешно исправлены.

С итогами тестирования можно ознакомиться на рисунке 22.

```
Request: POST http://localhost:8000/api/login
PASS API\AuthTest
✓ register
✓ register with error user exist
✓ login
✓ login with error incorrect data
Request: GET http://localhost:8000/api/builds
Request: POST http://localhost:8000/api/builds
Request: POST http://localhost:8000/api/builds/check
PASS API\BuildTest
✓ get builds
✓ create build
✓ build is not ready
Request: POST http://localhost:8000/api/components
Request: GET http://localhost:8000/api/components
Request: POST http://localhost:8000/api/components
Request: DELETE http://localhost:8000/api/components/4
Request: POST http://localhost:8000/api/components
PASS API\ComponentTest
✓ create component
✓ get components
✓ delete component
✓ create component without necessary attribute
Request: GET http://localhost:8000/api/users/me
Request: GET http://localhost:8000/api/users
Request: POST http://localhost:8000/api/user
PASS API\UserTest
✓ get auth user
✓ get users
✓ edit user
Tests: 14 passed
Time: 1.79s
```

Рисунок 21 – Результаты тестов

Также удалось выяснить процент покрытия кода тестами, результат составил около 52%, что является неплохим результатом.

A screenshot of a terminal window with a dark background. The prompt is 'PS D:\Projects\pc-configurator\backend>'. Below the prompt, there is a line of text 'Total Coverage 52.2 %' where the dots represent a progress bar. At the bottom of the terminal, there are icons for Git, TODO, Problems, Terminal, and Services.

```
PS D:\Projects\pc-configurator\backend>  
Total Coverage ..... 52.2 %  
PS D:\Projects\pc-configurator\backend>
```

Рисунок 22 – Процент покрытия кода тестами

В результате написанных тестов удалось обнаружить несколько неисправностей в работе приложения.

Также написанные тесты помогут в будущем сократить временные издержки при внедрении новых возможностей в систему, так как не будет необходимости проверять старые методы вручную заново, нужно лишь добавить новые тесты под реализованный функционал.

3.4 Разворачивание рабочего сервера

Разворачивание рабочего сервера производилось при помощи технологии VPS. Виртуальный сервер был выбран из-за необходимости в ручной настройке системного окружения, что, как правило, не могут позволить услуги обычного хостинга.

VPS – услуга предоставления в аренду так называемого виртуального выделенного сервера. В плане управления операционной системой по большей части она соответствует физическому выделенному серверу. В частности: root-доступ, собственные IP-адреса, порты, правила фильтрации и таблицы маршрутизации.

В качестве веб-сервера использовалось 2 виртуальных сервера Nginx. Один веб-сервер обеспечивает доступ к клиентской части приложения. Второй веб-сервер предоставляет доступ к API.

Данный подход обеспечивает легкость в обслуживании и масштабировании системы.

С конфигурацией веб-сервера можно ознакомиться в приложение 3.

По итогу работы имеем приложения работающее по внешнему ip-адресу <http://193.187.96.33/>

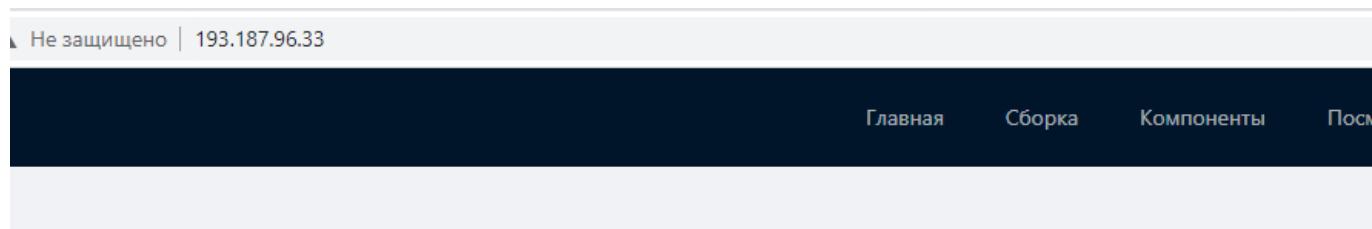


Рисунок 23 – Конфигуратор ПК развернутый на внешнем рабочем сервере

4 Демонстрация результатов

Ознакомиться с результатами работы можно на рисунках ниже.

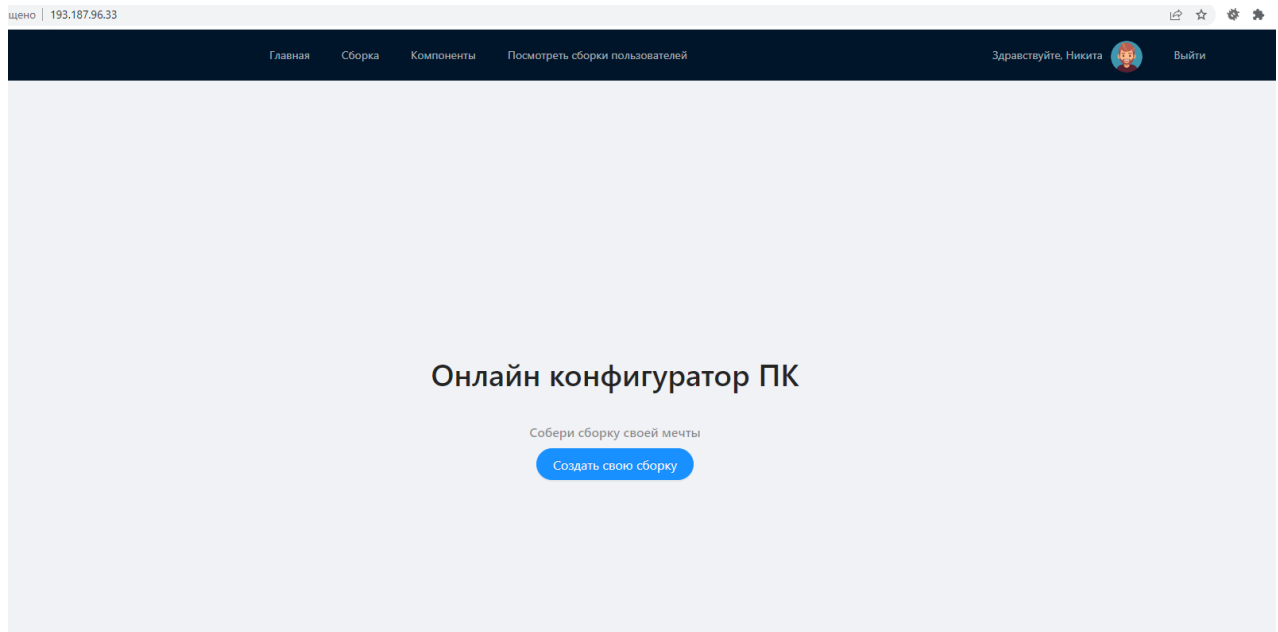


Рисунок 24 – Главная страница сайта

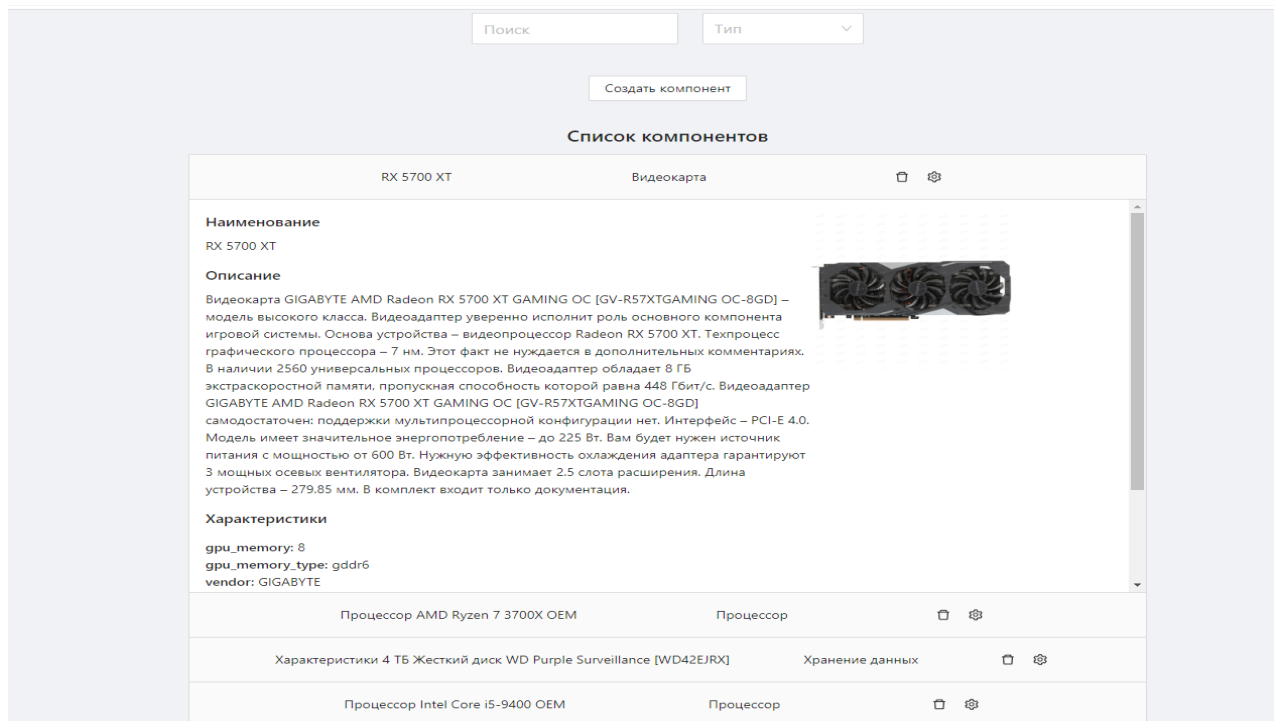


Рисунок 25 – Список компонентов

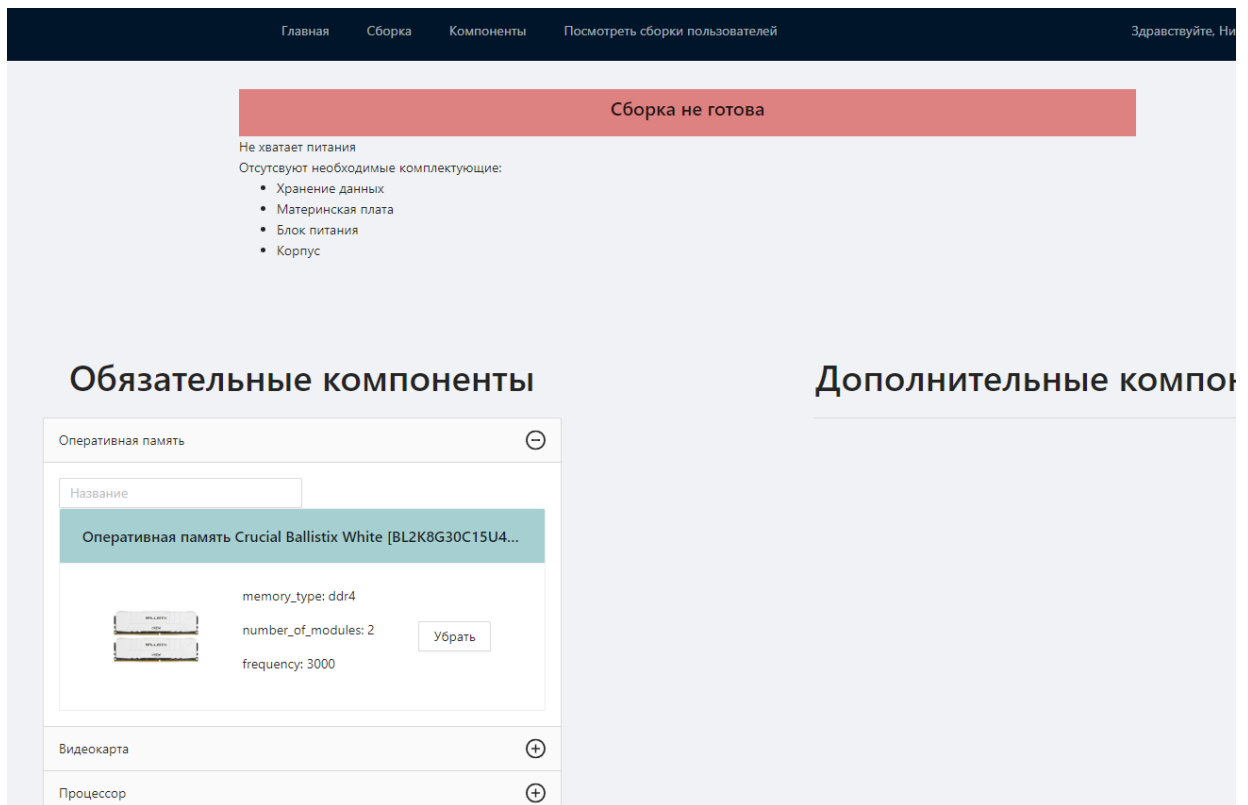


Рисунок 26 – Форма создания сборки

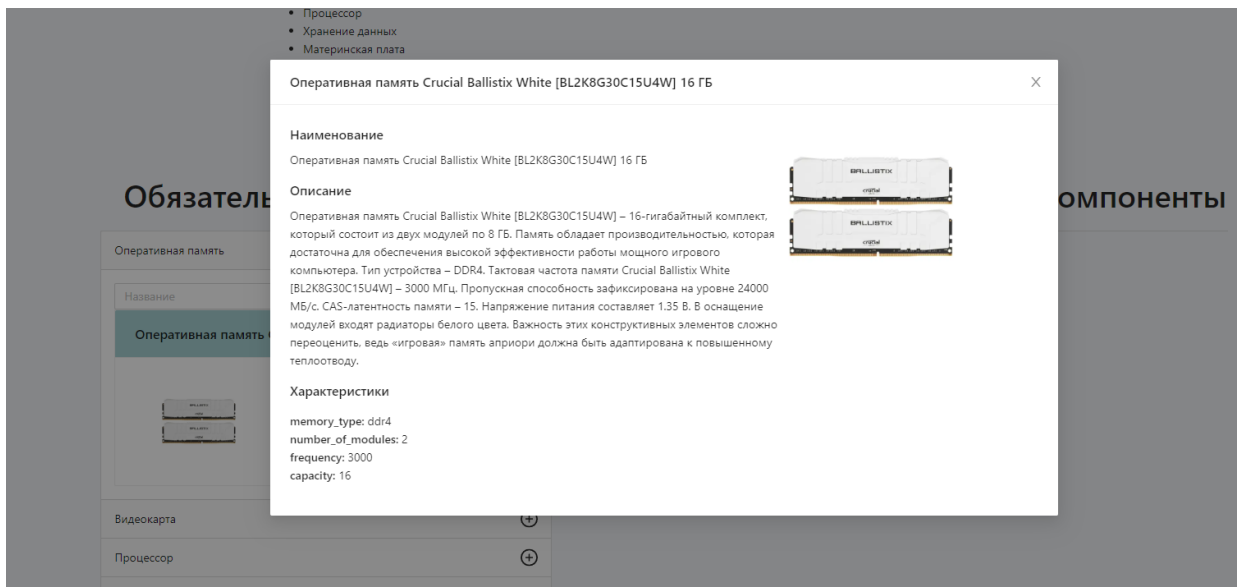









Рисунок 27 – Подробная информация о комплектующем




Моя лучшая сборка
 Сборка готова
 Оптимальная сборка для учебы
 Автор сборки: Никита alen2014@gmail.com

Удалить Изменить

Оперативная память Сл...	RX 5700 XT	Процессор AMD Ryzen 7 ...	Материнская плата GIGA...	Характеристики 4 ТБ Жес...	Корпус DEERCOOL MATRE...	Блок питания Chieftec Pr...
						

Бюджетная версия
 Сборка не готова
 Автор сборки: Никита alen2014@gmail.com

Удалить Изменить

Оперативная память Crucial Ballistix White [BL2K8G30C15U4W] 16 ГБ	Блок питания Chieftec Proton 850W [BDF-850C]	Процессор Intel Core i5-9400 OEM
		

Создать сборку

Рисунок 28 – Страница «Мои сборки»

**ЗАДАНИЕ ДЛЯ РАЗДЕЛА
«СОЦИАЛЬНАЯ ОТВЕТСТВЕННОСТЬ»**

Студенту:

Группа	ФИО
8К91	Лепейкин Никита Евгеньевич

Школа	ИШИТР	Отделение школы (НОЦ)	Отделение контроля и диагностики
Уровень образования	Бакалавриат	Направление/специальность	09.03.04 Программная инженерия

Тема ВКР:

Разработка приложения «Конфигуратор ПК»	
Исходные данные к разделу «Социальная ответственность»:	
<p>Введение</p> <ul style="list-style-type: none"> – <i>Характеристика объекта исследования (вещество, материал, прибор, алгоритм, методика) и области его применения.</i> – <i>Описание рабочей зоны (рабочего места) при разработке проектного решения</i> 	<p><i>Объект исследования:</i> веб-приложение «Конфигуратор ПК» для создания «виртуальныхборок» в режиме онлайн</p> <p><i>Область применения:</i> предоставление информации о готовности сборки ПК к эксплуатации</p> <p><i>Рабочая зона:</i> офис. Размеры помещения: 5*3 м.</p> <p><i>Количество и наименование оборудования рабочей зоны:</i> персональный компьютер – 1 шт.</p> <p><i>Рабочие процессы, связанные с объектом исследования, осуществляемые в рабочей зоне:</i> проектирование, разработка программного обеспечения.</p>
Перечень вопросов, подлежащих исследованию, проектированию и разработке:	
<p>1. Правовые и организационные вопросы обеспечения безопасности при разработке проектного решения:</p> <ul style="list-style-type: none"> – специальные (характерные при эксплуатации объекта исследования, проектируемой рабочей зоны) правовые нормы трудового законодательства; – организационные мероприятия при компоновке рабочей зоны. 	<p>«Трудовой кодекс Российской Федерации»</p> <p>ГОСТ 21889-76 Система «человек-машина». Кресло человека-оператора. Общие эргономические требования.</p> <p>ГОСТ Р 50923-96 Дисплеи. Рабочее место оператора. Общие эргономические требования и требования к производственной среде. Методы измерения</p>

<p>2. Производственная безопасность при разработке проектного решения:</p> <p>– Анализ выявленных вредных и опасных производственных факторов</p>	<p>Вредные факторы:</p> <ul style="list-style-type: none"> • Статические физические перегрузки, связанные срабочей позой. • Умственное перенапряжение, в том числе вызванноеинформационной нагрузкой. • Перенапряжение анализаторов, в том числе вызванноеинформационной нагрузкой. • Монотонность труда, вызывающая монотонию. • Отсутствие или недостаток необходимогоестественного освещения. • Отсутствие или недостаток необходимогоискусственного освещения. <p>Опасные факторы:</p> <ol style="list-style-type: none"> 1. Опасность поражения электрическим током 2. Требуемые средства коллективной защиты от выявленных факторов: системы естественного освещения, приборы искусственного освещения, очкизащитные со спектральными фильтрами ЛС и НСФ., устройства защитного отключения (УЗО), заземление, зануление корпусов электрооборудования.
<p>3. Экологическая безопасность при разработке проектного решения</p>	<p>Анализ негативного воздействия на окружающую природную среду: утилизация компьютеров и другой оргтехники</p> <p>Негативное воздействие на гидросферу и атмосферусовершается посредством производства различной оргтехники</p> <p>Воздействие на литосферу в результате образованияотходов при поломке оргтехники</p>
<p>4. Безопасность в чрезвычайных ситуациях при разработке проектногорешения</p>	<p>Возможные ЧС:</p> <p>Природные катастрофы (наводнения, цунами, ураган ит.д.);</p> <p>Геологические воздействия (землетрясения, оползни, обвалы, провалы территории);</p> <p>Техногенные аварии (пожар, сбои в электроснабжении)</p> <p>Наиболее типичная ЧС (пожар)</p>
<p>Дата выдачи задания для раздела по линейному графику</p>	<p>06.02.2023</p>

Задание выдал консультант:

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

Задание принял к исполнению студент:

Группа	ФИО	Подпись	Дата
8К91	Лепейкин Никита Евгеньвич		

5 Социальная ответственность

Введение

Объектом исследования являются веб-приложение для создания «виртуальных» сборок.

Выпускная квалификационная работа представляет собой проектирование и реализацию сервиса «Онлайн-конфигуратор персонального компьютера», позволяющего производить «виртуальную» сборку персонального компьютера с функцией автоматизации.

Область применения: предоставление информации о готовности сборки к эксплуатации. Сервис решает проблемы, связанные со сложностью самостоятельныхборок персонального компьютера ввиду отсутствия технических знаний об устройстве ЭВМ и многих тонкостях при сборке компьютера у рядового пользователя.

Целью разработанного сервиса является автоматизацией и облегчение процесса самостоятельных сборки персонального компьютера, информирования пользователя о текущем состоянии сборки ПК и готовности к эксплуатации.

Рабочий процесс в основном заключается в пребывании в офисе с размерами помещения: 5*3 м в положение сидя, занимаясь проектированием и непосредственно разработкой программного обеспечения, взаимодействуя с оборудованием таким как персональный компьютер.

В данном разделе выпускной квалификационной работы рассмотрены опасные и вредные факторы, оказывающие влияние на производственную деятельность программиста. Также рассмотрены вопросы техники пожарной безопасности, правила поведения во время чрезвычайных ситуаций и способы предотвращения их возникновения.

5.1 Правовые и организационные вопросы обеспечения безопасности

5.1.1 Правовые нормы трудового законодательства

Трудовые отношения между работодателем и работником регулируются с помощью законодательного акта «Трудовой кодекс Российской Федерации» [25]. В нём регламентируются права и обязанности работодателя и работника, вопросы организации труда, режим труда и отдыха, оплата и нормирование труда, компенсации работникам, защита персональных данных работника, урегулирование трудовых споров. Ниже приведены несколько наиболее важных положений:

- Нормальная продолжительность рабочего времени не должна превышать 40 часов в неделю;

- В течение рабочего дня (смены) работнику должен быть предоставлен перерыв для отдыха и питания продолжительностью не более двух часов и не менее 30 минут, который в рабочее время не включается. Правилами внутреннего трудового распорядка или трудовым договором может быть предусмотрено, что указанный перерыв может не предоставляться работнику, если установленная для него продолжительность ежедневной работы (смены) не превышает четырех часов (в ред. Федерального закона от 18.06.2017 N 125-ФЗ);
- Всем работникам предоставляются выходные дни (еженедельный непрерывный отдых).
- Эргономические требования к правильному расположению и компоновке рабочей зоны

Основными элементами рабочего места программиста являются: рабочий стол, рабочий стул (кресло), дисплей, клавиатура, мышь; вспомогательными – пюпитр, подставка для ног.

Согласно ГОСТ Р 50923-96 «Дисплеи. Рабочее место оператора. Общие эргономические требования и требования к производственной среде. Методы измерения» [2], рабочее место с дисплеем должно обеспечивать оператору возможность удобного выполнения работ в положении сидя и не создавать перегрузки костно-мышечной системы.

Регулируемая высота рабочей поверхности стола должна изменяться в пределах от 680 до 800 мм, а конструкция рабочего стола должна обеспечивать возможность размещения на рабочей поверхности необходимого комплекта оборудования и документов с учетом характера выполняемой работы. Размеры рабочей поверхности стола должны быть: глубина - не менее 600 мм, ширина - не менее 1200 мм.

Клавиатура должна иметь возможность свободного перемещения. Клавиатуру следует располагать на поверхности стола на расстоянии от 100 до 300 мм от переднего края, обращенного к оператору, или на специальной регулируемой по высоте рабочей поверхности, отделенной от основной столешницы.

Кресло оператора, может быть, с профилированными и непрофилированными элементами согласно ГОСТ 21889-76 «Система "Человек-машина". Кресло человека-оператора» [27]. Поверхность сиденья может быть плоской с наклоном 0-5°, или профилированной с углами наклона сиденья. Опорная плоскость сиденья может быть плоской или профилированной с радиусом кривизны поясничной опоры, равным 460 мм, радиусом изгиба для грудного отдела позвоночника, равным 620 мм и другими точками изгиба.

При размещении рабочих мест необходимо исключить возможность прямой засветки экрана источником естественного освещения.

При выполнении выпускной квалификационной работы правовых и организационных нарушений по указанным требованиям не было выявлено, рабочее место было оборудовано согласно всем нормам и правилам.

5.2 Производственная безопасность

В данном подразделе приведен анализ вредных и опасных факторов, которые воздействуют на разработчиков программного обеспечения, выполняющих работы на своих рабочих местах.

Все выявленные факторы приведены в таблице 3.

Таблица 4 – Возможные опасные и вредные производственные факторы на рабочем месте инженера-программиста

Факторы (ГОСТ 12.0.003-2015)	Нормативные документы
Статические физические перегрузки, связанные с рабочей позой.	ГОСТ 12.0.003-2015 «Система стандартов безопасности труда. Опасные и вредные факторы» [28]
Умственное перенапряжение, в том числе вызванное информационной нагрузкой.	МР 2.2.9.2311-07 «Профилактика стрессового состояния работников при различных видах профессиональной деятельности» [29]
Перенапряжение анализаторов, в том числе вызванное информационной нагрузкой.	МР 2.2.9.2311-07 «Профилактика стрессового состояния работников при различных видах профессиональной деятельности» [29]
Монотонность труда, вызывающая монотонию.	МР 2.2.9.2311-07 «Профилактика стрессового состояния работников при различных видах профессиональной деятельности» [29]
Отсутствие или недостаток необходимого естественного освещения.	СП 52.13330.2016 «Естественное искусственное освещение» [30]

Факторы (ГОСТ 12.0.003-2015)	Нормативные документы
Отсутствие или недостаток необходимого искусственного освещения.	СП 52.13330.2016 «Естественное искусственное освещение» [30]
Производственные факторы, связанные с электрическим током, вызываемым разницей электрических потенциалов, под действие которого попадает работающий.	ГОСТ Р 12.1.019-2017 «Система стандартов безопасности труда (ССБТ). Электробезопасность. Общие требования и номенклатура видов защиты» [31]

Из данной таблицы следует, что на разработчиков программного обеспечения в ходе их деятельности воздействуют только физические и психофизиологические факторы, а химические и биологические факторы отсутствуют.

5.2.1 Статические физические перегрузки

Работа программиста является малоподвижной деятельностью, а значит может вызывать гиподинамию. Как известно, ограничение физической активности приводит к нарушениям в работе опорно-двигательного аппарата, сердечно-сосудистой системы, желудочно-кишечного тракта. Все эти нарушения оказывают значительное влияние на состояние организма работника и негативно сказываются как на его повседневной жизни, так и на качестве его работы.

Для профилактики гиподинамии следует предпринимать следующие меры:

- увеличение физической активности;
- регулярные перерывы в работе для небольшой разминки.

5.2.2 Умственное перенапряжение и перенапряжение анализаторов, монотонность труда

Монотонный труд является частью практически любого процесса разработки, так как каждый специалист выполняет ряд каких-либо действий, связанных с его задачами. Кроме того, программисты по роду своей деятельности проводят много времени, решая сложные и нестандартные задачи и проблемы, из-за чего может возникать умственное перенапряжение (переутомление). Этот негативный фактор играет важную роль в эмоциональном состоянии. При умственном перенапряжении у сотрудника могут наблюдаться следующие негативные последствия:

- нарушение сна или сонливость;
- отсутствие или повышение аппетита;
- приступы тошноты, обмороки;
- головная боль;
- раздражительность, нервозность;
- усталость, апатия;
- медлительность;
- снижение внимательности и работоспособности.

Для снижения уровня умственного перенапряжения разработчика программного обеспечения следует принимать следующие меры:

- во время рабочего дня делать регулярные перерывы;
- вне рабочего времени проводить время на свежем воздухе;
- нормализовать режим сна;
- регулярно и сбалансировано питаться;
- выделять время в течении дня на разминку или на полноценные тренировки;
- стараться организовывать свое рабочее время таким образом, чтобы не заниматься одной задачей на протяжении длительного времени;
- крупные задачи делить на небольшие подзадачи.

Согласно МР 2.2.9.2311-07 [31], меры профилактики стрессовых состояний предусматривают внедрение рациональных режимов труда и отдыха, комплекса оздоровительно-профилактических мероприятий для предупреждения воздействия стресс-факторов на организм работающих.

При постоянном взаимодействии с компьютером (набор текстов, ввод данных и т.п.), при исключении возможности периодического переключения на другие виды трудовой деятельности, не связанные с ПЭВМ, рекомендуется организация перерывов по 10 - 15 мин. через каждые 45 - 60 мин. работы.

5.2.3 Отсутствие или недостаток необходимого искусственного освещения

Недостаточная освещенность рабочей зоны является вредным производственным фактором, приводящим к быстрому утомлению и снижению работоспособности человека на предприятии. При недостаточной освещенности помещения человек быстрее устает, снижается внимание и концентрация. Продолжительная работа в условиях низкой освещенности приводит к ухудшению зрения.

Нормы естественного, искусственного и совместного освещения регламентируются СП 52.13330.2016 «Естественное и искусственное освещение» [7]. Разработка программного обеспечения относится к работам высокой точности (наименьший или эквивалентный объект различения 0,30 – 0,50 мм), разряд Б, подразряд 1, относительная продолжительность зрительной работы при направлении зрения на рабочую поверхность не менее 70%.

В таблице 5 представлены требования к искусственному освещению рабочего помещения.

Таблица 5 – Требования к искусственному освещению рабочего помещения

Искусственное освещение			
Освещенность на рабочей поверхности от системы общего освещения, лк	Цилиндрическая освещенность, лк	Объединенный показатель UGR, не более	Коэффициент пульсации освещенности, Кп, %, не более
400	100	19	15

Нехватка естественного освещения или его полное отсутствие в течение рабочего дня может повлечь болезни психофизиологического характера, невротические расстройства, снизить работоспособность и концентрацию сотрудника.

Разработка программного обеспечения относится к работам высокой точности (наименьший или эквивалентный объект различения 0,30 – 0,50 мм), разряд Б, подразряд 1, относительная продолжительность зрительной работы при направлении зрения на рабочую поверхность не менее 70%. Требования к естественному освещению рабочего помещения согласно СП 52.13330.2016 «Естественное и искусственное освещение» [31] представлены в таблице 7.

Таблица 6 – Требования к естественному освещению рабочего помещения

Коэффициент естественной освещенности, %, при	
верхнем или комбинированном освещении	боковом освещении
3,0	1,0

При недостаточной освещенности помещения может помочь расширение оконных проемов и установка качественных источников искусственного освещения.

Искусственное освещение в помещениях для эксплуатации ПК должно осуществляться системой общего равномерного освещения. В случаях преимущественной работы с документами, следует применять системы комбинированного освещения (к общему освещению дополнительно устанавливаются светильники местного освещения).

5.2.4 Опасность поражения электрическим током

Работа программиста происходит в непосредственной близости от электрических сетей и приборов, поэтому работник должен с осторожностью обращаться с электропроводкой и компьютером, а также помнить об опасности поражения электрическим током.

Покрытие полов следует делать из однослойного линолеума, что снизит величины зарядов статического электричества. Несмотря на то, что эти величины безопасны для здоровья человека, вычислительная техника подвергается опасности при воздействии зарядов такого рода.

Поражение электрическим током является опасным производственным фактором. Вероятность поражения повышается при:

- повышенной влажности воздуха в помещении (более 75%);
- высокой температуре воздуха и поверхностей (более 35 °С);
- наличии токопроводящей пыли;
- неверной проектировке рабочего места;
- отсутствии защитных конструкций для проводов;
- наличии посторонних предметов на электроприборах.

Мерами защиты от воздействия электрического тока при неисправности изоляции являются защитное заземление, зануление [32] и использование устройств защитного отключения (УЗО).

5.3 Экологическая безопасность

Большое количество процессов, операций и материалов, используемых при изготовлении электронных средств, являются источниками огромного количества веществ, имеющих неблагоприятное воздействие на человека и биосферу. При изготовлении элементной базы, электронных изделий, при обработке, выращивании полупроводниковых кристаллов, при изготовлении интегральных схем, в процессе гальванического производства утилизация исходных материалов часто происходит с низким коэффициентом использования, огромное количество их идет в отходы, попадая в атмосферу, гидросферу, загрязняя почву. Таким образом, наряду с истощением природных запасов дефицитных материалов происходит

загрязнение окружающей среды, что ведет к губительным последствиям для отдельных экосистем и биосферы в целом.

На данном рабочем месте выявлен предполагаемый источник загрязнения окружающей среды, а именно воздействие на литосферу, гидросферу и атмосферу результате образования отходов при поломке предметов вычислительной техники и оргтехники.

Согласно ГОСТ Р 53692—2009, вышедшее из строя ПЭВМ и сопутствующая оргтехника относится к IV классу опасности и подлежит специальной утилизации.

Первым этапом является утилизация обезвреженных (инертных) отходов. Во время утилизации может быть произведена переработка бракованных или вышедших из употребления видов продукции, изделий, их составных частей и отходов от них путем разборки (разукрупнения), переплавки, использования других технологий с обеспечением рециркуляции(восстановления) органической и неорганической составляющих.

Вторым этапом является безопасное размещение отходов I—IV классов опасности на соответствующих полигонах или уничтожение [33].

5.4 Безопасность в чрезвычайных ситуациях

Возможные ЧС:

- Техногенные (взрывы, пожары, обрушение помещений);
- Природные (наводнения, ураганы, бури, природные пожары);
- Биологические (эпидемии, пандемии);
- Антропогенные (война, терроризм).

При работе с вычислительной техникой наиболее типичной чрезвычайной ситуацией является пожар, так как в современных ЭВМ очень высокая плотность размещения элементов электронных схем. К причинам электрического характера можно отнести короткое замыкание, искрение, статическое электричество.

Для предотвращения возникновения пожара необходимо:

- Регулярно проводить инструктажи по пожарной безопасности;
- Разместить в помещении план эвакуации и плакаты с краткой информацией о действиях, предпринимаемых при возникновении пожара;
- Соблюдать правила и нормы при монтаж электрических приборов и электрической проводки;
- Оборудовать помещение пожарной сигнализацией, а также средствами тушения пожара;

В случае возникновения пожара каждый сотрудник должен:

- незамедлительно сообщить об этом в пожарную охрану;
- принять меры по эвакуации людей, материальных ценностей согласно плану эвакуации;
- отключить электроэнергию, приступить к тушению пожара первичными средствами пожаротушения.

Возможный пожар на рабочем месте относится к классам А1, А2, Е [10]. Первичные средства пожаротушения являются: огнетушители порошковые переносные с порошками типа АВСЕ, огнетушители углекислотные.

5.5 Выводы по главе

В результате работы по главе «Социальная ответственность» были выявлены основные нормативные акты для обеспечения безопасности жизнедеятельности на рабочем месте, рассмотрены наиболее значимые опасные и вредные факторы, возникающие при проектировании и разработкесервиса для изучения иностранных языков, описано влияние процесса разработки программного обеспечения на окружающую среду и меры, необходимые для уменьшения влияния вредных и опасных факторов на организм человека и для сокращения негативного влияния процесса разработки программного обеспечения на окружающую среду.

Рабочее место выполнения выпускной квалификационной работы соответствует выявленным требованиям. Для обеспечения необходимого уровня освещения используются несколько энергосберегающих ламп и дополнительное настольное освещение, а уровень шума на рабочем месте находится в пределах допустимого диапазона. Для защиты от воздействия электрического тока проводится проверка состояния ПЭВМ и соблюдаются правила безопасности при работе с ней.

Согласно правилам установок электроустановок, рабочее помещение инженера-программиста относится к помещениям без повышенной опасности поражения электрическим током.

Согласно пункту 1.1.13 ПУЭ-7 рабочая зона является помещением безповышенной опасности. Согласно «Правилам по охране труда при эксплуатации электроустановок» [34] персонал должен иметь первую группу по электробезопасности.

Работа в офисе относится к категории тяжести труда Ia [35] – работы выполняются при оптимальных условиях внешней производственной среды и при оптимальной величине физической, умственной и нервно-эмоциональной нагрузки.

Рабочая зона относится к категории В по взрывопожарной и пожарнойопасности.

Эксплуатация ПЭВМ не оказывает значительного вреда экологии, однако неправильная утилизация комплектующих ПЭВМ и макулатуры может оказывать негативное воздействие на литосферу.

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

Студенту:

Группа	ФИО
8К91	Лепейкин Никита Евгеньевич

Школа	ИШИТР	Отделение школы (НОЦ)	Отделение контроля и диагностики
Уровень образования	Бакалавриат	Направление/специальность	Программная инженерия

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

1. <i>Стоимость ресурсов научного исследования (НИ): материально-технических, энергетических, финансовых, информационных и человеческих</i>	Оклад инженера – 21 760 руб. Оклад руководителя – 33 664 руб.
2. <i>Нормы и нормативы расходования ресурсов</i>	Премимальный коэффициент 30%; Коэффициент доплат и надбавок 20%; Районный коэффициент 30%; Коэффициент дополнительной заработной платы 12%; Накладные расходы 16%.
3. <i>Используемая система налогообложения, ставки налогов, отчислений, дисконтирования и кредитования</i>	Коэффициент отчислений на уплату внебюджетные фонды 27,1%

Перечень вопросов, подлежащих исследованию, проектированию и разработке:

1. <i>Оценка коммерческого потенциала, перспективности и альтернатив проведения НИ с позиции ресурсоэффективности и ресурсосбережения</i>	Анализ конкурентных технических решений
2. <i>Планирование и формирование бюджета научных исследований</i>	Формирование плана и графика разработки: -определение структуры и трудоемкости работ; -разработка графика Гантта. Формирование бюджета затрат: -затраты на специальное оборудование; -заработная плата (основная и дополнительная); -отчисления на социальные цели; -накладные расходы.

3. <i>Определение ресурсной (ресурсосберегающей), финансовой, бюджетной, социальной и экономической эффективности исследования</i>	<i>Определение потенциального эффекта исследования</i>
--	--

Перечень графического материала (с точным указанием обязательных чертежей):

2. <i>Оценка конкурентоспособности технических решений</i>
3. <i>Матрица SWOT</i>
4. <i>Альтернативы проведения НИ</i>
5. <i>График проведения и бюджет НИ</i>
6. <i>Оценка ресурсной, финансовой и экономической эффективности НИ</i>

Дата выдачи задания для раздела по линейному графику	06.02.2023
---	------------

Задание выдал консультант:

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

Задание принял к исполнению студент:

Группа	ФИО	Подпись	Дата
8К91	Лепейкин Никита Евгеньевич		

6 Финансовый менеджмент, ресурсоэффективность и ресурсосбережение

Разработка НИ производится группой работников, состоящей из двух человек – руководителя и студента.

Данная выпускная квалификационная работа заключается в проектировании и реализации сервиса для создания «виртуальных» сборок персонального компьютера (далее сборки ПК или просто сборки), проверки готовности сборки к эксплуатации, различные пояснение к состоянию текущей сборки.

Целью раздела «Финансовый менеджмент, ресурсоэффективность и ресурсосбережение» является определение перспективности и успешности НИ, оценка его эффективности, уровня возможных рисков, разработка механизма управления и сопровождения конкретных проектных решений на этапе реализации.

Для достижения обозначенной цели необходимо решить следующие задачи:

1. Оценить коммерческий потенциал и перспективность разработки НИ;
2. Осуществить планирование этапов выполнения исследования;
3. Рассчитать бюджет затрат на исследования;
4. Произвести оценку научно-технического уровня исследования и оценку рисков.

6.1 Потенциальные потребители результатов исследования

Целевой рынок – потребители младшего и среднего возраста, любой национальности, социальной принадлежности и профессии, с любым уровнем дохода, планирующие самостоятельно собрать персональный компьютер. В виду сложности самостоятельной сборки компьютера из-за низкого уровня знаний об устройстве ЭВМ у рядового пользователя данное приложения будет актуальной для указанной выше категории людей.

Наиболее значимыми критериями, которыми должен обладать потребитель и на которые будет нацелено приложение, является: желание собрать свой компьютер самостоятельно, но не имея должных знаний для того, чтобы собрать компьютер без каких-либо консультаций.

6.2 Анализ конкурентных технических решений

При рассмотрении различных сервисов для конструирования сборок ПК, в качестве главных конкурентных решений с похожими возможностями можно выделить

2 приложения: Конфигуратор ПК компании «DNS» и Конфигуратор ПК компании «Ситилинк».

На сайте интернет-магазина «DNS» имеется возможность сделать свою собственную сборку, а после ее можно купить в их физическом магазине (при наличии всех комплектующих в ассортименте).

Сайт имеет следующие возможности:

- Регистрация/Авторизация
- Удобный пользовательский интерфейс
- Подробная информация о товаре
- Проверка комплектующих на совместимость
- Возможность сохранить свою сборку и купить ее в их магазине
- Каждый товар имеет свой собственный рейтинг
- Поиск по категориям
- Фильтрация и сортировка
- Имеются пользовательские готовые сборки

В целом из недостатков можно выделить, то, что конфигуратор ПК компании продвигает покупку сборки в их магазине с помощью данного приложения, из-за этого невозможно добавить в сборку комплектующие, которых сейчас нет в наличии. Также нельзя добавить комплектующие, которые, как правило, устаревшие или являются труднодоступными в такого рода интернет-площадок.

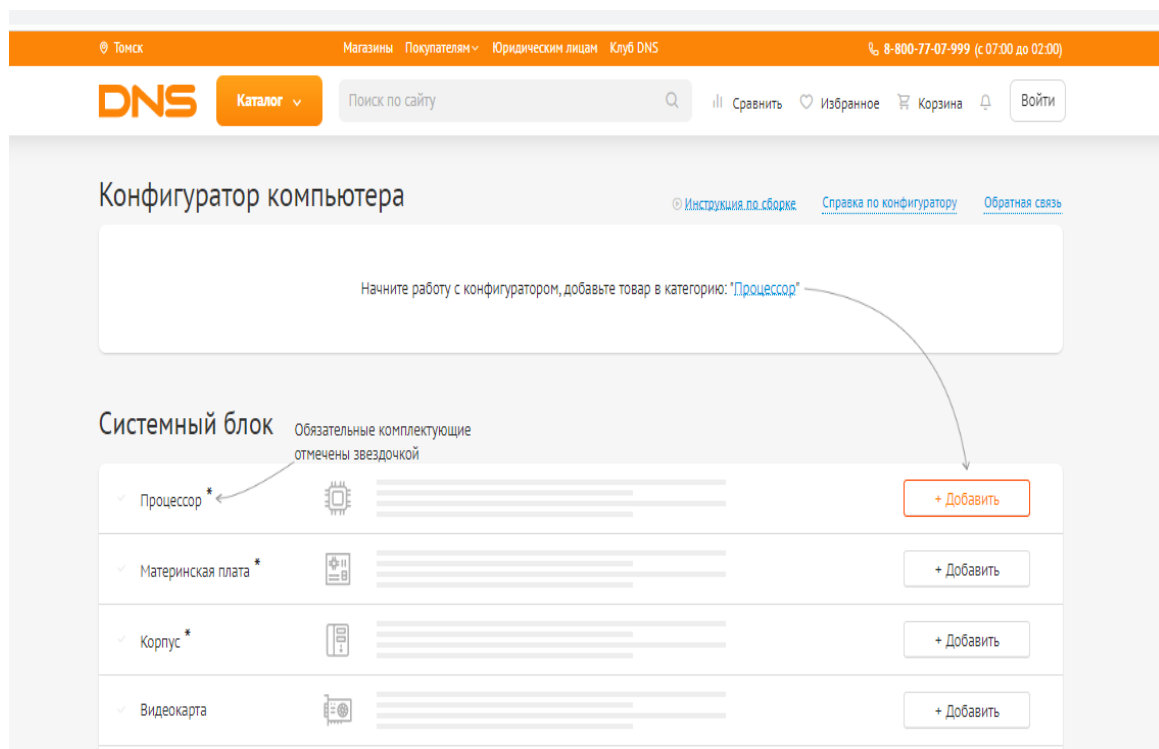


Рисунок 29 – Конфигуратор ПК «DNS»

Аналогично «DNS», «Ситилинк» также является российским интернет-магазином цифровой и бытовой техники с возможностью сборки ПК.

Сайт имеет следующие достоинства:

- Регистрация/Авторизация
- Удобный пользовательский интерфейс
- Пользовательские сборки ПК с выбором разделов (игровые, офисные, для работы и тд.)
- Проверка комплектующих на совместимость
- Возможность сохранить свою сборку и купить ее в их магазине
- Каждый товар имеет свой собственный рейтинг
- Поиск по категориям
- Фильтрация и сортировка
- Возможность сохранить свою сборку в формате Excel

При этом имея ровно те же недостатки, что и конфигуратор ПК компании «DNS»

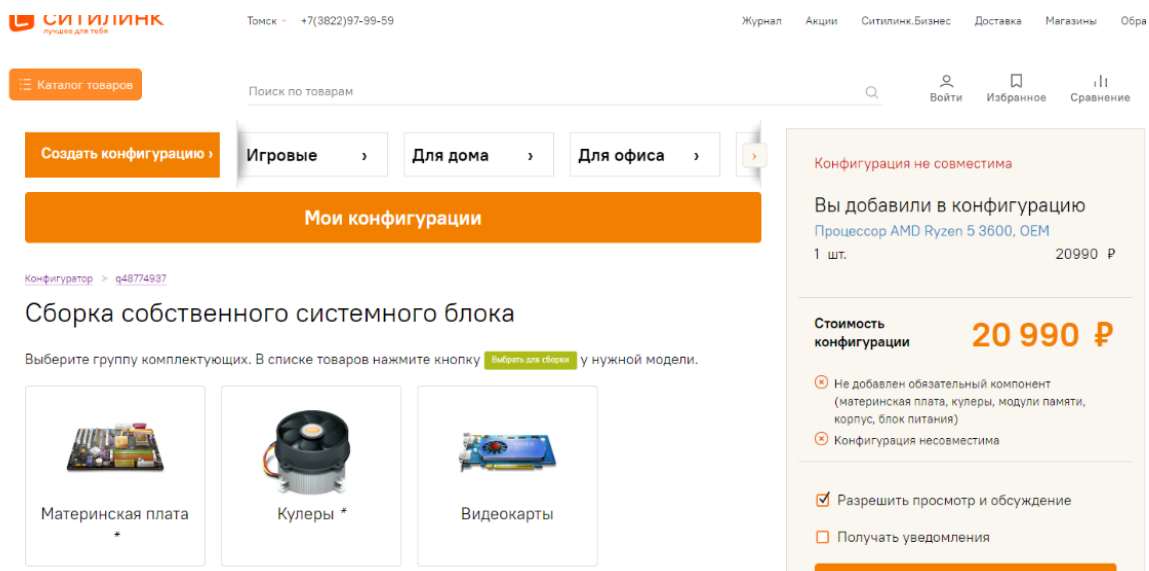


Рисунок 30 – Конфигуратор ПК «Ситилинк»

Анализ конкурентных технических решений был проведен помощью оценочной карты, представленной в таблице 8 (индекс Т отвечает за текущее решение).

Таблица 7 – Оценочная карта сравнения конкурентных технических решений

Критерии оценки	Вес критерия	Баллы			Конкурентоспособность		
		Бт	Бк _{DNS}	Бк _{СИТИЛИНК}	Кт	Кк _{DNS}	Кк _{СИТИЛИНК}
Простота и удобство в эксплуатации	0,1	5	5	3	0,4	0,6	0,4
Наличие редких комплектующих	0,25	5	1	1	0,6	0,1	0,18
Возможность формировать отчеты в виде Excel-файла	0,25	4	1	5	0,4	0,1	0,6
Подробный анализ работоспособности сборки	0,3	5	3	3	0,8	0,5	0,5
Возможность просматривать сборки других пользователей	0,1	4	5	5	0,4	0,4	0,4
Итого	1	23	15	17	4,65	2,4	3,2

Анализ конкурентных решений определяется по формуле 1:

$$K = \sum V_i \times B_i, \quad (1)$$

где

К – конкурентоспособность научной разработки или конкурента;

V_i – вес показателя (в долях единицы);

B_i – балл i -го показателя.

Основываясь на рассчитанных значениях, можно предположить, что подробный анализ работоспособности сборки и наличия редких экземпляров комплектующих формирует высокое конкурентное преимущество. Существующим альтернативам не хватает более подробного вывода результата о готовности сборки в эксплуатации. Только один из рассмотренных конкурентов имеет возможность генерировать отчеты о сборках в виде Excel-файла.

6.3 Основные и альтернативные способы проведения научных исследований

В таблице 9 представлены возможные варианты реализации разработки.

Таблица 8 – Морфологическая таблица

	1	2	3
А. Фреймворк для разработки клиентской части веб-сервиса	React JS	Vue JS	-
Б. Фреймворк для разработки серверной части веб-сервиса	PHP (Laravel)	PHP (Symfony)	JS (Nest JS)
В. Система управления базами данных	MySQL	PostgreSQL	MongoDB
Г. Среда разработки	PHPStorm	Visual Studio Code	Sublime Editor

Основным вариантом решения является А1Б1В1Г1. Выбор данного варианта решения обусловлен такими факторами, как наличие опыта разработки с использованием данных технологий, использование реляционной базы данных, а также высокой популярности среди сообщества разработчиков по работе с данными технологиями

Альтернативными вариантами решения являются:

- А2Б2В2Г1 – Позволит использовать более простой в использовании фреймворк на клиентской части, при этом фреймворк на серверной части должен повысить качество кода, при этом увеличится время на разработку.
- А1Б3В3Г2 – Данное решение является позволит уменьшить время на разработку, а также снизить ее стоимость.

6.4 Технологии 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.

Анализ имеющихся конкурентных продуктов необходимо проводить с достаточной регулярностью, так как рынок ИТ находится в постоянном и активном движении на сегодняшний день. Данный анализ позволяет производить изменения текущего внедрения, чтобы сделать его наиболее перспективным и конкурентоспособным в дальнейшем. Результат QuaD-анализа представлен в таблице 10.

Таблица 9 – Оценка критериев в соответствии с технологией QuaD

Критерии оценки	Вес критерия	Баллы	Максимальный балл	Относительное значение	Средневзвешенное значение
Простота и удобство в эксплуатации	0,1	95	100	0,95	9,5
Наличие редких комплектующих	0,25	85	100	0,85	21,25
Возможность формировать отчеты в виде Excel-файла	0,25	95	100	0,95	23,75

Подробный анализ работоспособности сборки	0,3	90	100	0,9	27
Возможность просматривать сборки других пользователей	0,1	85	100	0,85	8,5
Итого:					90

Средневзвешенное значение показателя качества и перспективности научной разработки равен 90 (попадает в промежуток 80–100), то есть такая разработка считается перспективной.

6.5 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).

Сильные (S) и слабые (W) стороны являются факторами внутренней среды объекта анализа, (то есть тем, на что сам объект способен повлиять); возможности (O) и угрозы (T) являются факторами внешней среды (то есть тем, что может повлиять на объект извне и при этом не контролируется объектом).

В рамках данного анализа выявлены сильные и слабые стороны проекта, а также его возможности и угрозы. Все параметры SWOT анализа данного проекта представлены в таблице 11.

Таблица 10 – Матрица SWOT-анализа

Сильные стороны	Возможности во внешней среде
<p>С1. Возможность добавлять редкие комплектующие</p> <p>С2. Просмотр сборок других пользователей</p> <p>С3. Подробный анализ о совместимости сборке и готовности к эксплуатации</p>	<p>В1. Увеличение пользовательских базы в соседних странах</p> <p>В2. Продажа готового продукта для крупной компании</p> <p>В3. Выведение продукта на мировой рынок</p> <p>В4. Приоритетная политика правительства РФ по поддержки ИТ сектора</p>
Слабые стороны	Угрозы внешней среды
<p>Сл1. Сложность поддержки системы</p> <p>Сл2. Необходимость длительной поддержки</p> <p>Сл3. Необходимость в ручном режиме добавлять новые комплектующие в систему</p> <p>Сл4. Недостаточная проработка дизайна приложения системы расширения.</p>	<p>У1. Некупаемость проекта в связи с низким количеством пользователей</p> <p>У2. Несоответствие ожиданиям клиента</p> <p>У3. Снижение спроса пользователей на покупку собственного ПК</p>

В дальнейшем выявить необходимости проведения стратегических изменений с помощью сопоставлению сильных и слабых сторон проекта с его угрозами и возможностями внешней среды.

Таблица 11 – Интерактивная матрица проекта с возможностями проекта

Возможности проекта	Сильные стороны			Слабые стороны			
	С1	С2	С3	Сл1	Сл2	Сл3	Сл4
В1	+	+	+	+	+	+	+
В2	+	-	+	+	+	-	-
В3	+	+	+	+	-	+	-
В4	-	-	-	0	+	0	+

Таблица 12 – Интерактивная матрица проекта с возможностями проекта

Угрозы проекта	Сильные стороны			Слабые стороны			
		С1	С2	С3	Сл1	Сл2	Сл3
У1	-	+	-	0	+	+	+
У2	+	+	+	0	-	+	+
У3	+	+	+	0	-	+	0

Анализ интерактивных матриц показал следующие корреляцию сильных и слабых сторон с возможностями и угрозами проекта:

V1C1C2C3; V2C1C3; V3C1C2C3;

V1Сл1Сл2Сл3Сл4; V2Сл1Сл2; V3Сл1Сл3; V4Сл2Сл4

У1С2; У2С1С2С3; У3С1С2С3

У1Сл2Сл3Сл4; У2Сл3Сл4; У3Сл3

Таким образом стало понятно какие факторы более тесно связаны друг с другом, а какие к каким вообще не имеют отношения, а также что самая большая угроза – это конкуренция на рынке и ситуация в мире, на чем и строится проект.

В результате можно построить итоговую таблица SWOT-анализа

Таблица 13 – Итоговая матрица SWOT-анализа проекта

		Внутренние факторы	
		Сильные стороны проекта: С1. Возможность добавлять редкие комплектующие С2. Просмотр сборок других пользователей С3. Подробный анализ о совместимости сборки и готовности к эксплуатации	Слабые стороны проекта: Сл1. Сложность поддержки системы Сл2. Необходимость длительной поддержки Сл3. Необходимость в ручном режиме добавлять новые комплектующие в систему Сл4. Недостаточная проработка дизайна приложения системы расширения.
Внешние факторы	<p>В1. Увеличение пользовательских базы в соседних странах</p> <p>В2. Продажа готового продукта для крупной компании</p> <p>В3. Выведение продукта на мировой рынок</p> <p>В4. Приоритетная политика правительства РФ по поддержке IT сектора</p>	<p>1. Востребованность как среди конечных потребителей, так и среди крупных организаций</p> <p>2. Значительные конкурентные преимущества</p>	<p>1. Полностью бесплатное использование может повысить количество лояльных клиентов и репутацию организации</p> <p>2. Дизайн всегда можно изменить</p>
	<p>Угрозы:</p> <p>У1. Неокупаемость проекта в связи с низким количеством пользователей</p> <p>У2. Несоответствие ожиданиям клиента</p> <p>У3. Снижение спроса пользователей на покупку собственного ПК</p>	<p>1. Интуитивно понятный дизайн, а также возможность его изменить может повысить посещаемость ресурса</p> <p>2. Увидев преимущества данного приложения, пользователи будут рассказывать знакомым с такими же интересами, что позволит привлечь новых пользователей</p>	<p>1. Малые вычислительные мощности серверов могут уменьшить посещаемость сайта</p> <p>2. Неблагоприятные изменения в гос. законах могут повлечь убытки (федеральный закон о персональных данных)</p> <p>3. Ухудшение отношений между странами могут стать причиной ухудшения распространения сервиса и монетизации</p>

SWOT-анализ показал, что риски, связанные с устранением несоответствий на всех этапах эксплуатации продукта, должны занимать ключевую роль в расчете выгоды предлагаемой разработки. Общая картина говорит о том, что, несмотря на угрозы и слабые

стороны проекта, разработка обладает конкурентным преимуществом и является перспективной.

6.6 Планирование научно-исследовательских работ

6.6.1 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 научный руководитель. Научный руководитель определяет цели и задачи для студента, направляет и контролирует работу его работу, оценивает результаты проделанной работы и дает рекомендации студенту. Студент полностью отвечает за выполняемую работу. В таблице 15 представлен перечень работ, а также распределение исполнителей по ним.

Таблица 14 – Перечень работ и распределение исполнителей

№ работы	Наименование работы	Исполнители работы
1	Выбор научного руководителя бакалаврской работы	Лепейкин Н. Е.
2	Составление и утверждение темы бакалаврской работы	Лепейкин Н. Е. Мыцко Е. А. Чердынцев Е. С.
3	Составление календарного плана-графика выполнения бакалаврской работы	Лепейкин Н. Е.
4	Подбор и изучение литературы по теме бакалаврской работы	Лепейкин Н. Е.
5	Анализ предметной области	Лепейкин Н. Е.
6	Проектирование информационной системы	Лепейкин Н. Е.
7	Разработка клиентского части приложения	Лепейкин Н. Е.
8	Настройка серверной части приложения	Лепейкин Н. Е.
9	Согласование выполненной работы с научным руководителем	Лепейкин Н. Е. Мыцко Е.А.
10	Выполнение других частей работы (финансовый менеджмент, социальная ответственность)	Лепейкин Н. Е.
11	Подведение итогов, оформление работы	Лепейкин Н. Е.

6.6.2 Определение трудоемкости выполнения работ

Как правило, трудовые затраты образуют основную часть стоимости исследования, поэтому важным этапом планирования научно- исследовательской деятельности является определение трудоемкости работ.

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 работ.

Трудоёмкость оценивается по следующей формуле 2:

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

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

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

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

После оценки ожидаемой трудоёмкости работ, производится определение продолжительности каждой работы в рабочих днях по формуле 3:

$$Tp_i = \frac{t_{ож\ i}}{Ч_i}, \quad (3)$$

где Tp_i – продолжительность одной работы, раб. дн

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

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

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:

$$T_{k_i} = Tp_i \times k_{кал}, \quad (4)$$

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

Tp_i – это продолжительность выполнения i -й работы в рабочих днях;

$k_{кал}$ – это коэффициент календарности, равный 1,48.

Коэффициент календарности $k_{кал}$ рассчитывается по формуле 5:

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

где $k_{кал}$ – коэффициент календарности;

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

$T_{\text{вых}}$ – 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 118 дней, коэффициент календарности равен $k_{\text{кал}} = 1,48$.

6.6.3 Разработка графика проведения научного исследования

Таблица 15 – Временные показатели проведения научного исследования

Наименование работы	Исполнители работы	Трудоемкость работ, чел-дни		Длительность работ, дни		
		tmin	tmax	tож	T _п	T _к
Выбор научного руководителя бакалаврской работы	Лепейкин Н. Е.	1	2	1,4	1	1
Составление и утверждение темы бакалаврской работы	Лепейкин Н. Е. Мыцко Е. А. Чердынцев Е. С	2	7	4,5	5	5
Составление календарного плана-графика выполнения бакалаврской работы	Лепейкин Н. Е.	1	2	1,4	1	1
Подбор и изучение литературы по теме бакалаврской работы	Лепейкин Н. Е.	5	10	7	4	6
Анализ предметной области	Лепейкин Н. Е.	5	10	7	7	10
Проектирование информационной системы	Лепейкин Н. Е.	8	14	10,4	10	15
Разработка клиентской части приложения	Лепейкин Н. Е.	20	30	24	24	36
Разработка серверной части приложения	Лепейкин Н. Е.	10	20	14	14	21
Согласование выполненной работы с научным руководителем	Лепейкин Н. Е. Мыцко Е.А.	2	7	4,5	5	5
Выполнение других частей работы (финансовый менеджмент, социальная	Лепейкин Н. Е.					

ответственность)		5	12	7,8	8	12
Подведение итогов, оформление работы	Лепейкин Н. Е.	8	14	10,4	10	15
Итого	Лепейкин Н. Е.	47	93	65,4	81	119
	Мыцко Е.А.	9	17	12,2	7	10
	Чердынцев Е. С.	1	3	1,8	1	1

На основе данных таблицы 16 был построен календарный план-график, представленный на рисунке 35.

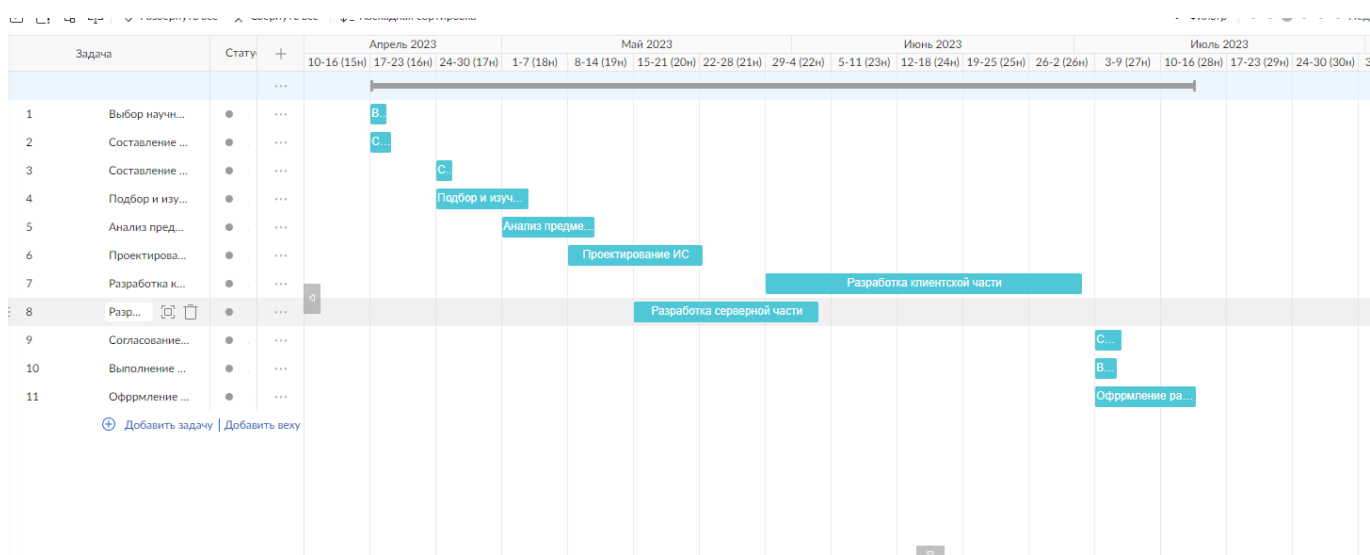


Рисунок 31 – Календарный план-график проведения работ

6.7 Бюджет научно-технического исследования

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

- расчет материальных затрат НТИ;
- затраты на специальное оборудование;
- основная заработная плата исполнителей;
- дополнительная заработная плата исполнителей;
- отчисления во внебюджетные фонды;
- накладные расходы.

6.7.1 Расчет материальных затрат НИИ

Заранее неизвестно какое количество электроэнергии и канцелярских товаров может понадобиться, поэтому в качестве материальных затрат ориентировочная сумма равная 5 тыс. рублям.

6.7.2 Расчет затрат на специальное оборудование для научных (экспериментальных целей)

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 один персональный компьютер, стоимостью 100 000 рублей.

Расчет амортизации ноутбука: первоначальная стоимость 100 000 рублей; срок полезного использования для машин офисных код 330.28.23.23 составляет 36 месяцев. Планируемое время использования для написания ВКР 6 месяцев.

Норма амортизация основных средств линейным способом рассчитывается по формуле б:

$$A_n = \frac{1}{n} * 100\%, \quad (6)$$

где n – установленный срок в месяцах;

A_n – норма амортизации

Тогда расчет амортизации компьютера

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

Ежемесячные амортизационные отчисления:

$$A_m = 100\,000 * 0,0278 = 2780 \text{ рублей}$$

Итоговая сумма амортизации ОС:

$$A = 2780 * 6 = 16680 \text{ рублей}$$

Таким образом, сумма затрат на специальное оборудование составляет 16680 рублей, в виде амортизационных отчислений.

6.7.3 Основная заработная плата исполнителей темы

Данная статья затрат включает основную заработную плату, премии и доплаты всех исполнителей проекта. В качестве исполнителей проекта выступают студенты и научный руководитель.

Зарботная плата рассчитывается по формуле 7:

$$Z_{\text{зп}} = Z_{\text{осн}} + Z_{\text{доп}}, \quad (7)$$

где $Z_{\text{зп}}$ – заработная плата исполнителя;

$Z_{\text{осн}}$ – основная заработная плата исполнителя;

$Z_{\text{доп}}$ – дополнительная заработная плата исполнителя (12%-15% от размера основной заработной платы).

Основную заработную плату можно получить по формуле 8:

$$Z_{\text{осн}} = Z_{\text{дн}} * T_{\text{р}} * (1 + K_{\text{пр}} + K_{\text{д}}) * K_{\text{р}}, \quad (8)$$

где, $Z_{\text{дн}}$ – среднедневная заработная плата, руб.

$K_{\text{пр}}$ – премиальный коэффициент (0,3);

$K_{\text{д}}$ – коэффициент доплат и надбавок (0,2-0,5);

$K_{\text{р}}$ – районный коэффициент (для Томска 1,3);

$T_{\text{р}}$ – продолжительность работ, выполняемых работником, раб. дни

Среднедневную заработную плату можно получить по формуле 9:

$$Z_{\text{дн}} = \frac{Z_{\text{м}} * M}{F_{\text{д}}}, \quad (9)$$

где $Z_{\text{м}}$ – месячный должностной оклад исполнителя, рубли;

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;

$F_{\text{д}}$ – действительный годовой фонд рабочего времени персонала по разработке.

Должностные оклады исполнителей проекта согласно приказу ТПУ представлены в таблице 17.

Таблица 16 – Месячные должностные оклады исполнителей

Исполнитель	Районный коэффициент (для Томска)	Размер месячного должностного оклада без учета коэффициента, рубли
Научный руководитель (должность – доцент, степень – кандидат технических наук)	1,3	33664
Студент (ассистент, без степени)	1,3	21760

Баланс рабочего времени для 6-дневной рабочей недели представлен в таблице 18.

Таблица 17 – Баланс рабочего времени (для 6-дневной недели)

Показатели рабочего времени	Дни
Календарные дни	365
Нерабочие дни (праздники/выходные)	118
Действительный годовой фонд рабочего времени	247

На основе формулы 8 и таблиц 17–18 была рассчитана среднедневная заработная плата:

$$Z_{\text{дн}} (\text{студент}) = \frac{21\,760 \cdot 10,4}{244} = 927,48 \text{ рублей}$$

$$Z_{\text{дн}} (\text{научный руководитель}) = \frac{33\,664 \cdot 10,4}{244} = 1434,86 \text{ рублей}$$

Расчет затрат на основную заработную плату приведен в таблице 19.

Таблица 18 – Расчет основной зарплаты исполнителей

№	Наименование этапов	Исполнители	Трудоемкость чел.-дн.	Зарботная плата, приходящаяся на один чел.-дн., тыс. руб.
1	Выбор научного руководителя бакалаврской работы	Лепейкин Н.Е	1	927,48
2	Составление и утверждение темы бакалаврской работы	Лепейкин Н.Е	1	927,48
		Мыцко Е.А.	5	7174,3
3	Составление календарного плана- графика выполнения бакалаврской работы	Лепейкин Н.Е	1	927,48
4	Подбор и изучение литературы по теме бакалаврской работы	Лепейкин Н.Е	6	5564,88
5	Анализ предметной области	Лепейкин Н.Е	10	9274,8
6	Проектирование информационной системы	Лепейкин Н.Е.	15	13912,2
7	Разработка клиентской части приложения	Лепейкин Н.Е.	36	33389,28
8	Разработка серверной части приложения	Лепейкин Н.Е.	21	19477,08
9	Согласование	Лепейкин Н.Е.	1	927,48

№ Наименование этапов		Исполнители	Трудоемкость чел.-дн.	Зарботная плата, приходящаяся на один чел.-дн., тыс. руб.
	выполненной работы с научным руководителем	Мыцко Е.А.	5	7174,3
10	Выполнение других частей работы (финансовый менеджмент, социальная ответственност ь)	Лепейкин Н.Е	12	11129,76
11	Подведение итогов, оформление работы	Лепейкин Н.Е	15	13912,2

Таблица 19 – Итоговый результат расчета основной зарплаты

Исполнители	Здн, руб.	Кпр	Кд	Кр	Тр	Зосн, руб.
Студент	927,48	0,3	0,2	1,3	119	110370,12
Научный руководитель	1434,86	0,3	0,2	1,3	10	14348,6
Итого:						124718,72

Итоговая сумма затрат на основную заработную плату составила 124718,72 руб.

6.7.4 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 10:

$$Z_{\text{доп}} = k_{\text{доп}} * Z_{\text{осн}}, \quad (10)$$

где $Z_{\text{доп}}$ – дополнительная заработная плата, рубли;

$k_{\text{доп}}$ – коэффициент дополнительной заработной платы (на стадии проектирования принимается равным 0,12 –0,2);

$Z_{\text{осн}}$ – основная заработная плата, рубли.

Расчет затрат на дополнительную заработную плату приведен в таблице 21.

Таблица 20 – Затраты на дополнительную заработную плату

Исполнители	$Z_{\text{осн}}$, руб.	$K_{\text{доп}}$	$Z_{\text{доп}}$, руб.
Студент	110370,12	0,12	13244,41
Научный руководитель	14348,6	0,12	2350,30
Итого:			15594,71

Итоговая сумма затрат на дополнительную заработную плату составила 15594,71 руб.

6.7.5 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 11.

$$Z_{\text{внеб}} = k_{\text{внеб}} * (Z_{\text{осн}} + Z_{\text{доп}}), \quad (11)$$

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

На 2014 г. в соответствии с Федеральным законом от 24.07.2009 №212-ФЗ установлен размер страховых взносов равный 30%. На основании пункта 1 ст.58 закона №212-ФЗ для учреждений, осуществляющих образовательную и научную деятельность в 2014 году, вводится пониженная ставка – 27,1%.

Расчет затрат на отчисления во внебюджетные фонды приведен в таблице 22.

Таблица 21 – Отчисления во внебюджетные фонды

Исполнители	Зосн, руб.	Здоп, руб.	Квнеб	Звнеб, руб.
Студент	110370,12	13244.41	0,271	33499.53
Научный руководитель	14348,6	2350,30	0,271	4525.40
Итого:				38024.93

Итоговая сумма отчислений во внебюджетные фонды составила 38024.93 руб.

6.7.6 Накладные расходы

Накладные расходы – расходы на организацию, управление и обслуживание процесса производства товара, оказания услуги; носят комплексный характер. Накладные расходы вычисляются по формуле 12:

$$Z_{\text{накл}} = \text{сумма статей}(1 - 5) * k_{\text{нр}}, \quad (12)$$

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

Расчет накладных расходов приведен в таблице 23.

Таблица 22 – Расчет накладных расходов

Статьи затрат	Сумма, руб.
Материальные затраты НТИ	5000
Затраты на специальное оборудование	16680
Затраты на основную заработную плату	124718,72
Затраты на дополнительную заработную плату исполнителям проекта	15594,71
Затраты на отчисления во внебюджетные фонды	38024,93
Коэффициент накладных расходов	0,16
Накладные расходы	32002,9376

Итоговая сумма накладных расходов составила 32002,93 руб.

6.7.7 Формирование бюджета затрат научно-исследовательского проекта

После того, как была подсчитана каждая из статей расходов, можно приступить к формированию общего бюджета затрат проекта. Итоговый бюджет затрат представлен в таблице 24.

Таблица 23 – Расчет бюджета затрат НТИ

Наименование статьи	Сумма, руб.	Удельный вес, %
Затраты на специальное оборудование	16680	5,6
Затраты на основную заработную плату	124718,72	55,5
Затраты на дополнительную заработную плату	15594,71	6,7
Страховые взносы	38024,93	18,8
Накладные расходы	32002,93	13,5
Общий бюджет	299362,42	100

Таким образом, общий бюджет НТИ составляет 188996,36 рублей.

6.8 Определение потенциального эффекта исследования

Для определения эффективности работы необходимо рассчитать интегральный показатель эффективности научного исследования, нахождение которого связано с определением финансовой эффективности и ресурсоэффективности. При расчёте показателей эффективности оценивались 2 варианта исполнения системы:

1. Клиентская часть реализуется на фреймворке React JS.
2. Клиентская часть реализуется на фреймворке Vue JS.

Стоимость разработки в этих двух вариантах является практически одинаковой, поэтому отдельный расчет для второго варианта не производился. Как следствие, интегральный финансовый показатель для обоих вариантов примем равным единице.

Существенным различием (и выгодным отличием первого варианта) является производительность и дальнейший потенциал развития первого варианта в будущем, в то время как второй вариант из-за технических особенностей вынужден будет столкнуться с ограниченной 80 производительностью, а дальнейшая поддержка, из-за падения интереса к фреймворку, повлечет более высокие затраты на дальнейшее развитие. Расчет интегральных показателей ресурсоэффективности обоих вариантов исполнения приведен в таблице 25.

Таблица 24 – Сравнительная оценка характеристик вариантов исполнения проекта

Объект исследования	Весовой коэффициент параметра	React JS	Vue JS
Критерий			
1. Гибкость настройки	0,2	5	4

Критерий \ Объект исследования	Весовой коэффициент параметра	React JS	Vue JS
2. Понятность и удобство пользовательского интерфейса	0,2	5	5
3. Скорость загрузки страницы сайта	0,2	4	3
4. Лёгкость масштабирования и дальнейшего развития	0,4	5	3
Итого	1	4,75	3,75

Сравнение интегрального показателя эффективности вариантов исполнения разработки позволит определить сравнительную эффективность проекта и выбрать наиболее целесообразный вариант из предложенных. Расчет приведен в таблице ниже:

Таблица 25 – Перечень работ и распределение исполнителей

№	Показатели	React JS	Vue JS
1	Интегральный финансовый показатель разработки	1	1
2	Интегральный показатель ресурсоэффективности разработки	4,75	3,75
3	Интегральный показатель эффективности	4,75	3,75
4	Сравнительная эффективность вариантов исполнения	1	0,79

Сравнив значения интегральных показателей эффективности, можно сделать вывод, что реализация программного обеспечения в первом исполнении является более эффективным вариантом решения задачи, 81 поставленной в данной работе с позиции финансовой и ресурсной эффективности. Исходя в качестве основного варианта исполнения проекта будет выбран 1 вариант (React JS).

6.9 Выводы по главе

В результате работы, проделанной по разделу «Финансовый менеджмент, ресурсоэффективность и ресурсосбережение», проект был исследован с экономической точки зрения.

В рамках данного раздела был проведен комплексный SWOT-анализ проекта, который позволил выявить его сильные и слабые стороны, а также определить соответствие его характеристик внешним факторам. Более того, были выявлены возможные пути дальнейшего развития и совершенствования проекта для повышения его конкурентоспособности. Также была произведена оценка качества и перспективности данного проекта.

Ещё одним ключевым моментом, рассмотренным в данном разделе, является планирование работ, выполняемых в рамках проекта. Составленный список необходимых работ с распределением исполнителей, а также вычисленная трудоемкость, позволили построить наглядный план-график работ в виде диаграммы Ганта. Это позволило рационально распорядиться временными ресурсами, отведенными на выполнение проекта.

Кроме того, был определен бюджет проекта. Определены затраты на заработные платы исполнителям с учетом страховых отчислений, рассчитаны накладные расходы, а также амортизация оборудования, задействованного в процессе выполнения проекта.

Сравнение интегральных показателей эффективности вариантов исполнения показывает, что наиболее выгодным с позиции финансовой и ресурсной эффективности является первый вариант исполнения (реализация клиентской части на фреймворке React JS), который и был реализован.

Заключение

В результате выполнения выпускной квалификационной работы получены теоретические и практические знания по проектированию программных систем. Удалось разработать веб-приложения с нуля имея только представление о решаемой проблеме.

В ходе работы был произведен обзор существующих аналогов, имеющих схожий функционал, выявлены достоинства и недостатки. Исходя из проведенного обзора была доказана необходимость разработки ввиду отсутствия приложений решающие рассмотренные ранее проблемы.

Выполнено проектирование системы, составлено техническое задание, дизайн, различные UML-диаграммы, а также ER-диаграмма предметной области.

В дальнейшем проведена программная реализация приложения, тестирование, а также разворачивание приложения на рабочем сервере.

Все задачи, которые были поставлены на время выполнения выпускной квалификационной работы были выполнены успешно.

Список использованных источников

1. Собрать компьютер или купить готовый: за и против [Электронный ресурс]. URL: <https://club.dns-shop.ru/blog/t-325-sborka-computera/21652-sobrat-komputer-vs-kupit-gotovyii-za-i-protiv/> (дата обращения: 08.04.2023).
2. Конфигуратор ПК: как собрать компьютер онлайн [Электронный ресурс]. URL: <https://expert.ru/2021/12/10/konfigurator-pk-kak-sobrat-kompyuter-onlayn/> (дата обращения: 10.04.2023).
3. Исикава К. Японские методы управления качеством. Сокр.пер. с англ. Под. Ред. А. В. Гличева. — М: Экономика, 1988. — 214 с.
4. Климчук В.А, Ядровская М. В. Анализ использования аппаратно-программных средств в учебной деятельности студентов [Электронный ресурс]. URL: <https://cyberleninka.ru/article/n/analiz-ispolzovaniya-apparatno-programmnyh-sredstv-v-uchebnoy-deyatelnosti-studentov> (дата обращения: 15.04.2023).
5. Конфигуратор ПК DNS URL: <https://www.dns-shop.ru/configurator/> (дата обращения: 16.04.2023).
6. Конфигуратор ПК Citilink [Электронный ресурс]. URL: <https://www.citilink.ru/configurator/> (дата обращения: 22.04.2023).
7. Конфигуратор ПК EDELWEISS [Электронный ресурс]. URL: <https://edelws.ru/constructor/> (дата обращения: 22.04.2023).
8. Конфигуратор ПК Регард. URL: <https://www.regard.ru/configurations> (дата обращения: 22.04.2023).
9. Конфигуратор ПК Xcom Shop URL: <https://www.xcom-shop.ru/pc-configurator/> (дата обращения: 22.04.2023).
10. Что такое PHP? URL: <https://blog.skillfactory.ru/glossary/php/> (дата обращения: 24.12.2021).
11. Язык программирования PHP: рейтинг, сферы применения, прогнозы экспертов URL: <https://ru.hexlet.io/blog/posts/zachem-izuchat-php-reyting-perspektivy-sfery-primeneniya> (дата обращения: 24.04.2023).
12. Руководство по PostgreSQL URL: <https://metanit.com/sql/postgresql/> (дата обращения: 25.04.2023).
13. Дэвид Флэнаган. 13.4.1. Букмарклеты // JavaScript. Подробное руководство = JavaScript. The Definite Guide / Перевод А. Киселева. — 5-е изд. — СПб.: «Символ-Плюс», 2008. — С. 267. — ISBN 5-93286-103.

14. Хвостенко Т. М., Великсар Д. С. Figma - перспективный инструмент современного веб-дизайнера // Вестник образовательного консорциума Среднерусский университет. Информационные технологии. — 2019. — № 2 (14). — С. 7—10.
15. Диаграмма вариантов использования URL: <https://www.kaznu.kz/content/files/pages/folder23404/%D0%9B%D0%B5%D0%BA%D1%86%D0%B8%D1%8F%205.pdf> (дата обращения: 25.04.2023).
16. Диаграммы деятельности (activity diagram). Основные объекты диаграммы URL: <https://ru.hexlet.io/blog/posts/zachem-izuchat-php-reyting-perspektivy-sfery-primeneniya> (дата обращения: 26.04.2023).
17. Построение диаграммы классов URL: https://flexberry.github.io/ru/gpg_class-diagram.html (дата обращения: 26.04.2023).
18. Основы правил проектирования базы данных [Электронный ресурс]. URL: <https://habr.com/ru/post/514364/> (дата обращения 26.04.2023)
19. Что такое RESTful API? [Электронный ресурс]. URL: <https://aws.amazon.com/ru/what-is/restful-api/> (дата обращения 01.05.2023)
20. Стандартный глоссарий терминов, используемых в тестировании программного обеспечения, версия 2.3, под ред. Erik van Veenendaal // International Software Testing Qualifications Board (ISTQB), 2014
21. Что такое Black Box Testing? [Электронный ресурс]. URL: <https://habr.com/ru/companies/ruvds/articles/700858/> (дата обращения 08.04.2023).
22. Основы PHPUnit [Электронный ресурс]. URL: https://klisl.com/phpunit_basics_1.html (дата обращения 01.05.2023).
23. Laravel Testing [Электронный ресурс]. URL: <https://laravel.com/docs/10.x/testing> (дата обращения 01.05.2023).
24. Что такое VDS? VPS - что это? [Электронный ресурс]. URL: <https://firstvds.ru/technology/chto-takoe-vds-vps-chto-eto> (дата обращения 02.05.2023).
25. «Трудовой кодекс Российской Федерации» от 30.12.2001 N 197-ФЗ (ред. от 25.03.2022).
26. ГОСТ Р 50923-96 «Дисплеи. Рабочее место оператора. Общие эргономические требования и требования к производственной среде. Методы измерения» [Электронный ресурс]. – Режим доступа: <https://docs.cntd.ru/document/1200025975> (дата обращения: 25.03.2022).
27. ГОСТ 21889-76 «Система "Человек-машина". Кресло человека- оператора» [Электронный ресурс]. – Режим доступа: <https://docs.cntd.ru/document/1200012832> (дата обращения: 25.03.2022).

28. ГОСТ 12.0.003-2015 «Система стандартов безопасности труда. Опасные и вредные факторы» [Электронный ресурс]. – Режим доступа: <https://docs.cntd.ru/document/1200136071> (дата обращения: 25.03.2022).
29. МР 2.2.9.2311-07 «Профилактика стрессового состояния работников при различных видах профессиональной деятельности» [Электронный ресурс]. – Режим доступа: <https://docs.cntd.ru/document/1200072234> (дата обращения: 25.03.2022).
30. СП 52.13330.2016 «Естественное и искусственное освещение» [Электронный ресурс]. – Режим доступа: <https://docs.cntd.ru/document/456054197> (дата обращения: 25.03.2022).
31. ГОСТ Р 12.1.019-2017 «Система стандартов безопасности труда (ССБТ). Электробезопасность. Общие требования и номенклатура видов защиты» [Электронный ресурс]. – Режим доступа: <https://docs.cntd.ru/document/1200161238> (дата обращения: 25.03.2022).
32. ГОСТ Р 53692-2009 «Ресурсосбережение. Обращение с отходами. Этапы технологического цикла отходов» [Электронный ресурс]. – Режим доступа: <https://docs.cntd.ru/document/1200081740> (дата обращения: 25.03.2022).
33. Федеральный закон от 22.07.2008 N 123-ФЗ (ред. от 30.04.2021) «Технический регламент о требованиях пожарной безопасности» [Электронный ресурс]. – Режим доступа: <https://docs.cntd.ru/document/902111644> (дата обращения: 25.03.2022).
34. Приказ Минтруда России от 15 декабря 2020 года N 903н «Об утверждении правил по охране труда при эксплуатации электроустановок» [Электронный ресурс]. – Режим доступа: <https://docs.cntd.ru/document/573264184> (дата обращения: 25.03.2022).
35. Приказ Минтруда России от 24.01.2014 N 33н (ред. от 27.04.2020) «Об утверждении Методики проведения специальной оценки условий труда, Классификатора вредных и (или) опасных производственных факторов, формы отчета о проведении специальной оценки условий труда и инструкции по ее заполнению» [Электронный ресурс]. – Режим доступа: http://www.consultant.ru/document/cons_doc_LAW_158398/ (дата обращения: 25.03.2022).

Приложение 1

Диаграмма классов

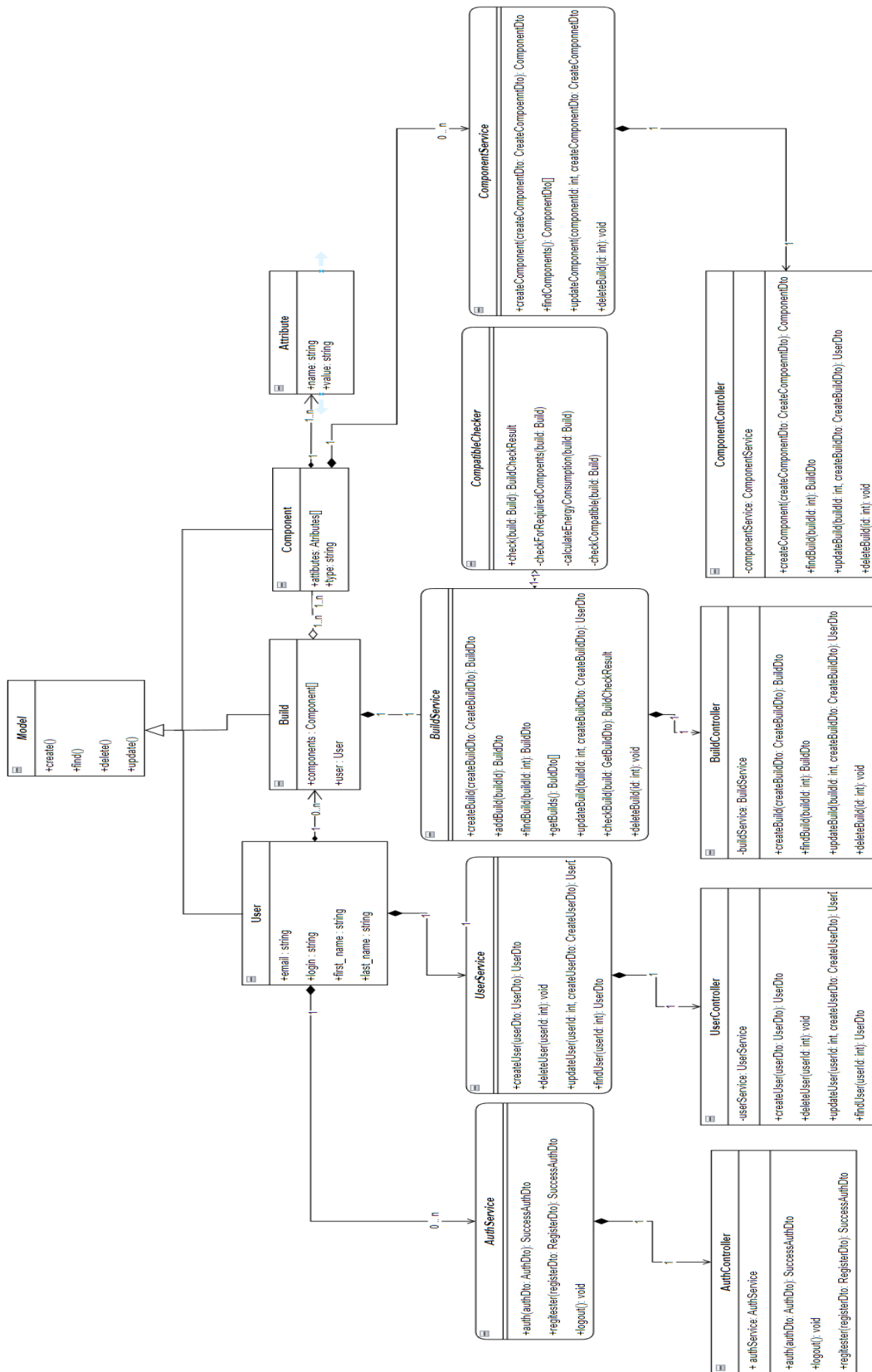


Рисунок 32. Диаграмма классов

Приложение 2

Класс APITest

```
<?php

namespace API;

use App\Models\User\User;
use Illuminate\Foundation\Testing\RefreshDatabase;
use Tests\TestCase;

abstract class APITest extends TestCase
{
    use RefreshDatabase;

    protected $seed = true;

    protected function setUp(): void
    {
        parent::setUp();

        $this->withHeaders([
            'Accept' => 'application/json'
        ]);
    }
}
```

```
public function asAdmin(): static
{
    $token = $this->createAdminToken();

    $this->withHeaders(['Authorization' => "Bearer $token"]);

    return $this;
}
```

```
public function asUser(): static
{
    $token = $this->createUserToken();

    $this->withHeaders(['Authorization' => "Bearer $token"]);

    return $this;
}
```

```
public function createUserToken()
{
    $user = User::updateOrCreate([
        'email' => 'user@test.ru',
    ], [
        'first_name' => 'user',
        'last_name' => 'user',
        'password' => '123123'
    ]);

    $user->roles()->create(['name' => 'user']);
}
```

```

        return $user->createToken('accessToken')->plainTextToken;
    }

    public function createAdminToken()
    {
        $user = User::updateOrCreate([
            'email' => 'admin@test.ru',
        ], [
            'first_name' => 'admin',
            'last_name' => 'admin',
            'password' => '123123'
        ]);

        $user->roles()->create(['name' => 'admin']);

        return $user->createToken('accessToken')->plainTextToken;
    }
}

```

Класс UserTest

```
<?php
```

```
namespace API;
```

```
use App\Http\Controllers\User\dto\EditUserDto;
```

```
use App\Http\Controllers\User\dto\GetUserDto;
```

```
use Illuminate\Testing\Fluent\AssertableJson;
```

```
class UserTest extends APITest
```

```
{
```

```
    public function testGetAuthUser(): void
```

```
    {
```

```
        $response = $this->asUser()->get('api/users/me');
```

```
        $response->assertStatus(200)->assertJson(fn(AssertableJson $json) => $json->hasAll(getClassProperties(GetUserDto::class)));
```

```
    }
```

```
    public function testGetUsers(): void
```

```
    {
```

```
        $response = $this->asUser()->get('api/users');
```

```
        $response->assertStatus(200)->assertJson(
```

```
            fn(AssertableJson $json) => $json->first(
```

```
                fn(AssertableJson $json) => $json->hasAll(getClassProperties(GetUserDto::class))
```

```
            ));
```

```
    }
```

```
    public function testEditUser()
```

```
    {
```

```
$updateUserDto = new EditUserDto(null, 'Nick', null, null);

$response = $this->asUser()->post('api/user', (array)$updateUserDto);

$response->assertStatus(200)->assertJson(
    fn(AssertableJson $json) => $json->hasAll(getClassProperties(GetUserDto::class))-
    >assertJson(['firstName' => 'Nick']
);
}
}
```


Приложение 3

Конфигурация backend сервера nginx

```
server {  
  
    listen 81 default_server;  
  
    listen [::]:81 default_server;  
  
    add_header X-Frame-Options "SAMEORIGIN";  
  
    add_header X-Content-Type-Options "nosniff";  
  
  
  
  
  
    # SSL configuration  
  
    #  
  
    # listen 443 ssl default_server;  
  
    # listen [::]:443 ssl default_server;  
  
    #  
  
    # Note: You should disable gzip for SSL traffic.  
  
    # See: https://bugs.debian.org/773332  
  
    #  
  
    # Read up on ssl_ciphers to ensure a secure configuration.  
  
    # See: https://bugs.debian.org/765782  
  
    #  
  
    # Self signed certs generated by the ssl-cert package  
  
    # Don't use them in a production server!  
  
    #
```

```
# include snippets/snakeoil.conf;

root /var/www/html/backend/public;

# Add index.php to the list if you are using PHP
index index.php;

charset utf-8;

server_name _;

location / {
    # First attempt to serve request as file, then
    # as directory, then fall back to displaying a 404.
    try_files $uri $uri/ /index.php?$query_string;
}

# pass PHP scripts to FastCGI server
#
location ~ \.php$ {
    include snippets/fastcgi-php.conf;

    # With php-fpm (or other unix sockets):
    fastcgi_pass unix:/run/php/php8.1-fpm.sock;
    # With php-cgi (or other tcp sockets):
```



```
#fastcgi_pass 127.0.0.1:9000;
}

# deny access to .htaccess files, if Apache's document root
# concurs with nginx's one

location ~ /\.ht {
    deny all;
}
}
```