

Министерство науки и высшего образования Российской Федерации  
 федеральное государственное автономное  
 образовательное учреждение высшего образования  
 «Национальный исследовательский Томский политехнический университет» (ТПУ)

Школа информационных технологий и робототехники  
 Направление подготовки 09.04.01 «Информатика и вычислительная техника»  
 Отделение школы (НОЦ) информационных технологий

### МАГИСТЕРСКАЯ ДИССЕРТАЦИЯ

Тема работы
<b>Проектирование и программная реализация web-сервиса кейтеринговых услуг для сотрудников компании ООО "Умный мир"</b>

УДК 004.774:004.455.1:794

Студент

Группа	ФИО	Подпись	Дата
8ВМ81	Гузиков Даниил Родионович		

Руководитель ВКР

Должность	ФИО	Ученая степень, звание	Подпись	Дата
Доцент	Цапко С.Г.	К.Т.Н.		

### КОНСУЛЬТАНТЫ ПО РАЗДЕЛАМ:

По разделу «Финансовый менеджмент, ресурсоэффективность и ресурсосбережение»

Должность	ФИО	Ученая степень, звание	Подпись	Дата
Доцент	Конотопский В.Ю.	К.Э.Н.		

По разделу «Социальная ответственность»

Должность	ФИО	Ученая степень, звание	Подпись	Дата
Доцент	Горбенко М.В.	К.Т.Н.		

### ДОПУСТИТЬ К ЗАЩИТЕ:

Руководитель ООП	ФИО	Ученая степень, звание	Подпись	Дата
Доцент	Кочегурова Е.А.	К.Т.Н.		

## ПЛАНИРУЕМЫЕ РЕЗУЛЬТАТЫ ОБУЧЕНИЯ

Код результата	Результат обучения (выпускник должен быть готов)
	<b>Общепрофессиональные компетенции</b>
P1	Воспринимать и самостоятельно приобретать, развивать и применять математические, естественнонаучные, социальноэкономические и профессиональные знания для решения нестандартных задач, в том числе в новой или незнакомой среде и в междисциплинарном контексте.
P2	Владеть и применять методы и средства получения, хранения, переработки и трансляции информации посредством современных компьютерных технологий, в том числе в глобальных компьютерных сетях
P3	Демонстрировать культуру мышления, способность выстраивать логику рассуждений и высказываний, основанных на интерпретации данных, интегрированных из разных областей науки и техники, выносить суждения на основании неполных данных, анализировать профессиональную информацию, выделять в ней главное, структурировать, оформлять и представлять в виде аналитических обзоров с обоснованными выводами и рекомендациями.
P4	Анализировать и оценивать уровни своих компетенций в сочетании со способностью и готовностью к саморегулированию дальнейшего образования и профессиональной мобильности. Владеть, по крайней мере, одним из иностранных языков на уровне социального и профессионального общения, применять специальную лексику и профессиональную терминологию языка.
	<b>Профессиональные компетенции</b>
P5	Выполнять инновационные инженерные проекты по разработке аппаратных и программных средств автоматизированных систем различного назначения с использованием современных методов проектирования, систем автоматизированного проектирования, передового опыта разработки конкурентно способных изделий.
P6	Планировать и проводить теоретические и экспериментальные исследования в области проектирования аппаратных и программных средств автоматизированных систем с использованием новейших достижений науки и техники, передового отечественного и зарубежного опыта. Критически оценивать полученные данные и делать выводы.
P7	Осуществлять авторское сопровождение процессов проектирования, внедрения и эксплуатации аппаратных и программных средств автоматизированных систем различного назначения.
	<b>Общекультурные компетенции</b>
P8	Использовать на практике умения и навыки в организации исследовательских, проектных работ и профессиональной эксплуатации современного оборудования и приборов, в управлении коллективом.
P9	Осуществлять коммуникации в профессиональной среде и в обществе в целом, активно владеть иностранным языком, разрабатывать документацию, презентовать и защищать результаты инновационной инженерной деятельности, в том числе на иностранном языке.
P10	Совершенствовать и развивать свой интеллектуальный и общекультурный уровень. Проявлять инициативу, в том числе в ситуациях риска, брать на себя всю полноту ответственности.

P11	Демонстрировать способность к самостоятельному обучению новым методам исследования, к изменению научного и научно-производственного профиля своей профессиональной деятельности, способность самостоятельно приобретать с помощью информационных технологий и использовать в практической деятельности новые знания и умения, в том числе в новых областях знаний, непосредственно не связанных со сферой деятельности, способность к педагогической деятельности.
-----	--

Министерство науки и высшего образования Российской Федерации  
 федеральное государственное автономное  
 образовательное учреждение высшего образования  
 «Национальный исследовательский Томский политехнический университет» (ТПУ)

Школа Информационных технологий и робототехники  
 Направление подготовки (специальность) 09.04.01 Информатика и вычислительная техника  
 Отделение школы (НОЦ) Информационных технологий

УТВЕРЖДАЮ:  
 Руководитель ООП

\_\_\_\_\_  
 (Подпись)    (Дата)    (Ф.И.О.)

**ЗАДАНИЕ**  
**на выполнение выпускной квалификационной работы**

В форме:

Магистерской диссертации
--------------------------

(бакалаврской работы, дипломного проекта/работы, магистерской диссертации)

Студенту:

Группа	ФИО
8ВМ81	Гузикову Даниилу Родионовичу

Тема работы:

Проектирование и программная реализация web-сервиса кейтеринговых услуг для сотрудников компании ООО "Умный мир"
--

Утверждена приказом директора (дата, номер)	133-30/с от 12.05.2020
---	------------------------

Срок сдачи студентом выполненной работы:	01.06.2020
--	------------

**ТЕХНИЧЕСКОЕ ЗАДАНИЕ:**

<b>Исходные данные к работе</b>	1.    Требование реализовать web-сервис кейтеринговых услуг
---------------------------------	---

<b>Перечень подлежащих исследованию, проектированию и разработке вопросов</b>	<ol style="list-style-type: none"> <li>1. Исследование предметной области</li> <li>2. Проектирование и реализация программных модулей</li> <li>3. Финансовый менеджмент, ресурсоэффективность и ресурсосбережения</li> <li>4. Социальная ответственность</li> <li>5. Заключения по работе</li> </ol>
<b>Перечень графического материала</b>	Презентация в формате *.pptx

**Консультанты по разделам выпускной квалификационной работы**

Раздел	Консультант
Финансовый менеджмент, ресурсоэффективность и ресурсосбережение	Конотопский В.Ю.
Социальная ответственность	Горбенко М.В.

**Названия разделов, которые должны быть написаны на русском и иностранном языках:**

Исследование предметной области

Проектирование и реализация программных модулей

<b>Дата выдачи задания по линейному графику</b>	10.03.2020
---	------------

**Задание выдал руководитель:**

Должность	ФИО	Ученая степень, звание	Подпись	Дата
Доцент	Цапко С.Г.	К.Т.Н.		

**Задание принял к исполнению студент:**

Группа	ФИО	Подпись	Дата
8ВМ81	Гузиков Даниил Родионович		

Министерство науки и высшего образования Российской Федерации  
 федеральное государственное автономное  
 образовательное учреждение высшего образования  
 «Национальный исследовательский Томский политехнический университет» (ТПУ)

Школа Информационных технологий и робототехники  
 Направление подготовки (специальность) 09.04.01 Информатика и вычислительная техника  
 Уровень образования Магистратура  
 Отделение школы (НОЦ) Информационных технологий  
 Период выполнения \_\_\_\_\_ (осенний / весенний семестр 2019 /2020 учебного года)

Форма представления работы:

Магистерская диссертация

(бакалаврская работа, дипломный проект/работа, магистерская диссертация)

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

Срок сдачи студентом выполненной работы:	01.06.2020
--	------------

Дата контроля	Название раздела (модуля) / вид работы (исследования)	Максимальный балл раздела (модуля)
29.05.2020	Основная часть	75
29.05.2020	Финансовый менеджмент, ресурсоэффективность и ресурсосбережение	15
29.05.2020	Социальная ответственность	10

**СОСТАВИЛ:**

**Руководитель ВКР**

Должность	ФИО	Ученая степень, звание	Подпись	Дата
Доцент	Цапко С.Г.	К.Т.Н.		

**СОГЛАСОВАНО:**

**Руководитель ООП**

Должность	ФИО	Ученая степень, звание	Подпись	Дата
Доцент	Кочегурова Е.А.	К.Т.Н.		

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

Студенту:

<b>Группа</b>	<b>ФИО</b>
8ВМ81	Гузикову Даниилу Родионовичу

<b>Школа</b>	Инженерная школа информационных технологий и робототехники	<b>Отделение школы (НОЦ)</b>	Отделение информационных технологий
<b>Уровень образования</b>	Магистратура	<b>Направление/специальность</b>	09.04.01 Информатика и вычислительная техника

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

1. <i>Стоимость ресурсов научного исследования (НИИ): материально-технических, энергетических, финансовых, информационных и человеческих</i>	Использовать действующие ценники и договорные цены на потребленные материальные и информационные ресурсы, а также указанную в МУ величину тарифа на эл. энергию
2. <i>Нормы и нормативы расходования ресурсов</i>	—
3. <i>Используемая система налогообложения, ставки налогов, отчислений, дисконтирования и кредитования</i>	Действующие ставки единого социального налога и НДС, ставка дисконтирования = 0,1 (см. МУ)

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

1. <i>Оценка коммерческого и инновационного потенциала НТИ</i>	Дать характеристику существующих и потенциальных потребителей (покупателей) результатов ВКР, ожидаемых масштабов их использования
2. <i>Разработка устава научно-технического проекта</i>	Разработать проект такого устава в случае, если для реализации результатов ВКР необходимо создание отдельной организации или отдельного структурного подразделения внутри существующей организации
3. <i>Планирование процесса управления НТИ: структура и график проведения, бюджет, риски и организация закупок</i>	Построение плана-графика выполнения ВКР, составление соответствующей сметы затрат, расчет цены результата ВКР.
4. <i>Определение ресурсной, финансовой, экономической эффективности</i>	Оценка экономической эффективности использования результатов ВКР, характеристика других видов эффекта

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

1. «Портрет» потребителя результатов НТИ
2. Сегментирование рынка
3. Оценка конкурентоспособности технических решений
4. Диаграмма FAST
5. Матрица SWOT
6. График проведения и бюджет НТИ - выполнить
7. Оценка ресурсной, финансовой и экономической эффективности НТИ - выполнить
8. Потенциальные риски

**Дата выдачи задания для раздела по линейному графику**

**Задание выдал консультант:**

<b>Должность</b>	<b>ФИО</b>	<b>Ученая степень, звание</b>	<b>Подпись</b>	<b>Дата</b>
Доцент	Конотопский В.Ю.	к.э.н.		

**Задание принял к исполнению студент:**

<b>Группа</b>	<b>ФИО</b>	<b>Подпись</b>	<b>Дата</b>
8ВМ81	Гузиков Даниил Родионович		

## ЗАДАНИЕ ДЛЯ РАЗДЕЛА «СОЦИАЛЬНАЯ ОТВЕТСТВЕННОСТЬ»

Студенту:

<b>Группа</b>	<b>ФИО</b>
8BM81	Гузикову Даниилу Родионовичу

<b>Школа</b>	Инженерная школа информационных технологий и робототехники	<b>Отделение (НОЦ)</b>	Отделение информационных технологий
<b>Уровень образования</b>	Магистратура	<b>Направление/специальность</b>	09.04.01 Информатика и вычислительная техника

Тема ВКР:

Проектирование и программная реализация web-сервиса кейтеринговых услуг для сотрудников компании ООО "Умный мир"	
<b>Исходные данные к разделу «Социальная ответственность»:</b>	
1. Характеристика объекта исследования (вещество, материал, прибор, алгоритм, методика, рабочая зона) и области его применения	Рабочая зона оборудована 12 местами, каждое из которых включает в себя компьютер с периферийными устройствами, расположенном на столе, стул. Технологический процесс представляет собой работу с языком программирования Javascript, в редакторе Visual Studio Code
Перечень вопросов, подлежащих исследованию, проектированию и разработке:	
<b>1. Правовые и организационные вопросы обеспечения безопасности:</b> <ul style="list-style-type: none"> <li>– специальные (характерные при эксплуатации объекта исследования, проектируемой рабочей зоны) правовые нормы трудового законодательства;</li> <li>– организационные мероприятия при компоновке рабочей зоны.</li> </ul>	<ul style="list-style-type: none"> <li>– СанПиН 2.2.2/2.4.1340 – 03</li> <li>– ГОСТ 12.2.032-78</li> <li>– Закон Томской области от 9 июля 2003 года № 83</li> <li>-ОЗ Об охране труда в Томской области (с изменениями на 4 июля 2014 года).</li> <li>– Трудовой кодекс Российской Федерации от 30.12.2001 N 197-ФЗ (ред. от 24.04.2020)</li> </ul>
<b>2. Производственная безопасность:</b> 2.1. Анализ выявленных вредных и опасных факторов 2.2. Обоснование мероприятий по снижению воздействия	<ul style="list-style-type: none"> <li>– Отклонение показателей микроклимата;</li> <li>– Недостаточная освещенность рабочей зоны;</li> <li>– Превышение уровня шума;</li> <li>– Опасные и вредные производственные факторы, связанные с электромагнитными полями;</li> <li>– Психофизиологические факторы;</li> <li>– Повышенное значение напряжения в электрической цепи, замыкание которой может произойти через тело человека;</li> </ul>
<b>3. Экологическая безопасность:</b>	– Анализ воздействия объекта на литосферу: утилизация отходов, связанные с выходом из строя ПК, люминесцентных ламп и др.
<b>4. Безопасность в чрезвычайных ситуациях:</b>	Типичная ЧС – пожар.



	<ul style="list-style-type: none"> <li>– разработка превентивных мер по предупреждению ЧС;</li> <li>– разработка действий в результате возникшей ЧС и мер по ликвидации её последствий.</li> </ul>
--	--

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

**Задание выдал консультант:**

Должность	ФИО	Ученая степень, звание	Подпись	Дата
доцент ООД ШБИП	Горбенко Михаил Владимирович	к.т.н.		

**Задание принял к исполнению студент:**

Группа	ФИО	Подпись	Дата
8ВМ81	Гузиков Даниил Родионович		

## РЕФЕРАТ

Выпускная квалификационная работа содержит: 93 страницы, 16 рисунков, 14 таблиц, 27 источников, 2 приложения.

Ключевые слова: web-сервис, кейтеринговые услуги, мобильное приложение, веб-приложение.

Объектом исследования является разрабатываемый web-сервис кейтеринговых услуг для сотрудников компании ООО «Умный мир».

Целью работы является проектирование и разработка web-сервиса кейтеринговых услуг для сотрудников компании ООО «Умный мир».

В процессе разработки был проведен анализ актуальных технологий для разработки веб-приложений.

В результате исследования был спроектирован и разработан web-сервис кейтеринговых услуг для сотрудников компании ООО «Умный мир».

Область применения: веб-сервис ориентирован на сотрудников компании ООО «Умный мир».

Основные характеристики: серверная часть приложения разработана на языке программирования PHP, клиентская часть приложения разработана на языке JavaScript, для оформления web-страниц использовались HTML и CSS, для хранения данных пользователей и журнала погружений использовалась СУБД PostgreSQL. Web-сервис предназначен для работы в любом современном веб-браузере и в мобильном приложении на IOS и Android.

В будущем планируется дальнейшая разработка и совершенствование web-сервиса в целях расширения функционала и улучшения интерфейса.

## ОБОЗНАЧЕНИЯ И СОКРАЩЕНИЯ

СУБД – Система Управления Базой Данных

SQL – Structured Query Language

JS – язык программирования JavaScript

Web-сервис – программа, которая организывает взаимодействие между сайтами

API – Application Programming Interface

PHP – язык программирования для серверной стороны веб-сайтов

Фреймворк - программное обеспечение, облегчающее разработку и объединение разных компонентов большого программного проекта

MVC – model-view-controller

CRUD – create-read-update-delete

HTML – язык гипертекстовой разметки

CSS – каскадные таблицы стилей

SCSS - это метаязык на основе CSS, предназначенный для увеличения уровня абстракции CSS кода и упрощения файлов каскадных таблиц стилей

DOM – это независимый от платформы и языка программный интерфейс, позволяющий программам и скриптам получить доступ к содержимому HTML документов, а также изменять содержимое, структуру и оформление таких документов

XML – расширяемый язык разметки

JSX – препроцессор, который добавляет синтаксис XML к JavaScript

## ОГЛАВЛЕНИЕ

ОБОЗНАЧЕНИЯ И СОКРАЩЕНИЯ .....	11
ВВЕДЕНИЕ .....	144
1. Исследование предметной области .....	155
1.1 Описание предметной области .....	155
1.2 Краткий обзор архитектуры web-сервиса кейтеринговых услуг и используемых технологий.....	155
2. Проектирование и реализация программных модулей .....	188
2.1 Требования и постановка задач .....	188
2.2 Выбор программных средств, технологии программирования для серверной части .....	19
2.3 Выбор программных средств, технологии программирования для веб-клиента .....	20
2.4 Выбор программных средств, технологии программирования для мобильного приложения.....	22
2.5 Управление глобальным состоянием в клиентских приложениях .....	24
2.6 Работа с Git .....	25
2.7 Организация хранения данных.....	26
2.8 Диаграмма вариантов использования web-сервиса.....	28
2.9 Диаграмма последовательности .....	29
2.10 Диаграмма деятельности «Оформление заказа» .....	30
2.11 Раздел авторизации.....	33
2.12 Раздел «Меню».....	34
2.13 Раздел «Профиль».....	37
2.14 Раздел «Корзина» .....	38
3. Финансовый менеджмент, ресурсоэффективность и ресурсосбережения.....	41

3.1 Оценка коммерческого потенциала и перспективности проведения научных исследований с позиции ресурсоэффективности и ресурсосбережения .....	41
3.2 Планирование проектных работ.....	44
3.3 Определение ресурсной (ресурсосберегающей), финансовой, бюджетной, социальной и экономической эффективности исследования .	54
Выводы по разделу .....	57
4. Социальная ответственность.....	59
Введение .....	59
4.1 Правовые и организационные вопросы обеспечения безопасности .....	59
4.2 Производственная безопасность .....	61
4.3 Экологическая безопасность .....	74
4.4 Безопасность в чрезвычайных ситуациях .....	75
ЗАКЛЮЧЕНИЕ .....	77
Conclusion .....	78
Список источников .....	79
Приложение А .....	82
Приложение Б.....	92

## ВВЕДЕНИЕ

В последнее время заметен рост оказания кейтеринговых услуг на территории России. Множество компаний по общественному питанию организуют доставку еды в офис или в квартиру жителям множества городов.

Компанией ООО «Умный мир» заключен договор со службой доставки «Александра», г. Томск, на доставку обедов сотрудникам. В настоящее время сотрудники заходят на сайт компании «Александра», выбирают там себе блюда и переносят выбранные блюда в google-таблицу, по которой затем офис-менеджер делает общий заказ на всех сотрудников в каждый из будних дней.

Актуальность разработки web-сервиса кейтеринговых услуг состоит в автоматизации процесса заказа сотрудниками компании обеда, что значительно сократит время заказа и облегчит работу офис-менеджера, которому не нужно будет заниматься этим вопросом.

Заказчиком web-сервиса является компания ООО «Умный мир».

В процессе выполнения работы были решены следующие задачи:

- выбор технологий для разработки web-сервиса;
- проектирование web-сервиса;
- разработка веб-приложения и мобильного приложения;
- разработка серверной части приложения;
- организация хранения данных.

## **1. Исследование предметной области**

### **1.1 Описание предметной области**

Термин «Кейтеринг» происходит от английского слова «cater», что означает «поставка провизии». Производное слово «catering» обозначает общественное питание. Под этим термином понимают, как оказание услуг по организации питания сотрудников подрядными компаниями, так и ресторанное обслуживание за пределами стационарных помещений, а также организация питания на различных выездных мероприятиях. На практике кейтеринг означает не только доставку и приготовление еды. В это понятие также входят обслуживание, оформление помещений, сервировка столов и иные подобные услуги [7].

Компенсация компанией расходов сотрудников на питание выгодны не только сотрудникам, но и работодателю, так как сотрудники компании более мотивированы на плодотворную работу, следовательно, повышается производительность труда и ценность компании на рынке труда возрастает.

### **1.2 Краткий обзор архитектуры web-сервиса кейтеринговых услуг и используемых технологий**

Было решено использовать для web-сервиса трехуровневую архитектуру, которая включает в себя следующие уровни:

- Клиент: этот уровень не должен иметь прямых связей с базой данных (по требованиям безопасности и масштабируемости), быть нагруженным основной бизнес-логикой (по требованиям масштабируемости) и хранить состояние приложения (по требованиям надёжности). На этот уровень обычно выносятся только простейшая бизнес-логика: интерфейс авторизации, алгоритмы шифрования, проверка вводимых значений на допустимость и соответствие формату, несложные операции с данными (сортировка, группировка, подсчёт значений), уже загруженными на терминал.

- Сервер приложений: располагается на втором уровне, на нём сосредоточена основная часть бизнес-логики. Вне его остаются только фрагменты, экспортируемые на клиента (терминалы), а также элементы логики, относящиеся к базе данных (хранимые процедуры и триггеры). Реализация данного компонента обеспечивается связующим программным обеспечением. Серверы приложений проектируются таким образом, чтобы добавление к ним дополнительных экземпляров обеспечивало горизонтальное масштабирование производительности программного комплекса и не требовало внесения изменений в программный код приложения.
- Сервер баз данных: обеспечивает хранение данных и выносится на отдельный уровень, реализуется, как правило, средствами систем управления базами данных, подключение к этому компоненту обеспечивается только с уровня сервера приложений.

Используемая трехуровневая архитектура web-сервиса схематично изображена на рисунке 1.

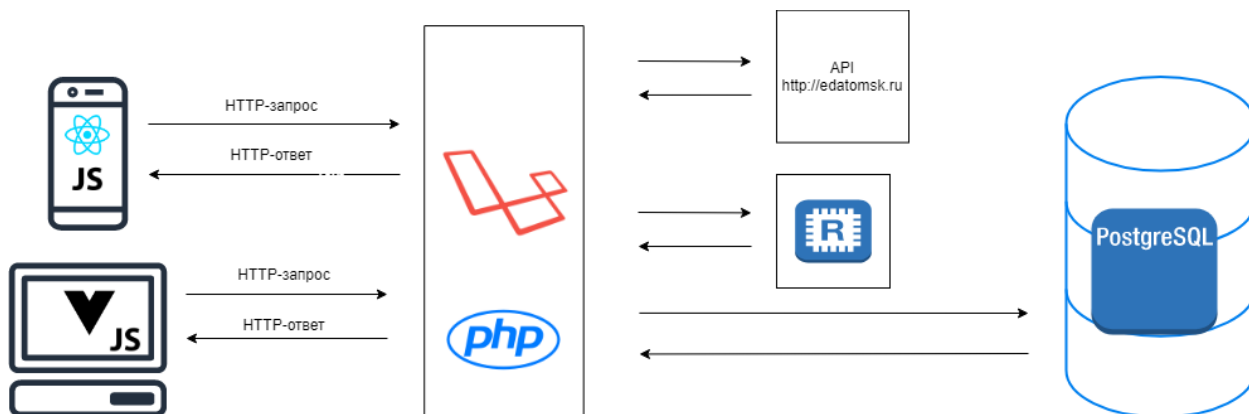


Рисунок 1 – Используемая архитектура

Для организации хранения данных использовалась СУБД PostgreSQL.

PostgreSQL - это универсальная и объектно-реляционная система управления базами данных с открытым исходным кодом. PostgreSQL была разработана на основе POSTGRES 4.2 в Беркли, университет Калифорнии [9].

Изначально идея создания PostgreSQL была в использовании его на UNIX-подобных платформах. Однако впоследствии PostgreSQL была также



разработана и для других платформ, таких как Mac OS X, Solaris и Windows. Данная СУБД является бесплатной, что позволяет свободно использовать, модифицировать и распространять PostgreSQL в любой форме.

PostgreSQL требует минимальных усилий из-за его стабильности. Следовательно, разработка приложения на основе PostgreSQL уменьшит общую стоимость владения в сравнении с другими системами управления базами данных.

Для разработки веб-сервера использовался язык PHP, являющийся серверным языком скриптов. Обычно используется для разработки статических или динамических веб-сайтов и веб-приложений. Данный язык имеет множество фреймворков, таких как Symfony, Laravel, Yii2, Zend. Было решено остановиться на Laravel, описание возможностей которого будет в разделе 2.2.

Также веб-сервер взаимодействует с API службы доставки Александра для получения актуального меню.

В качестве сервера для хранения очередей использовался Redis. Использование очереди позволяет отложить на какой-то период задачу, обычно это применяется для отправки e-mail или генерации pdf-файлов пользователям.

Для разработки веб-клиента и мобильного приложения был выбран объектно-ориентированный язык программирования JavaScript. На этом языке также имеется множество фреймворков для разработки веб-приложений и мобильных приложений, например, для веб-приложений пользуются популярностью React JS, Vue JS, Angular и т.д., описание которых будет в разделе 2.3. Для того, чтобы frontend-разработчики могли также разрабатывать интерфейсы для мобильных приложений, были созданы React native и Vue native, описание возможностей которых будет в разделе 2.4.

## **2. Проектирование и реализация программных модулей**

### **2.1 Требования и постановка задач**

Разрабатываемый web-сервис кейтеринговых услуг должен удовлетворять следующим требованиям:

- Web-сервис должен получать данные об актуальном меню в службе доставки «Александра» на неделю вперёд.
- В компании каждый сотрудник может заполнить корзину с дневным лимитом в 200 рублей. В случае его превышения, нужно уведомить об этом пользователя.
- Мобильное приложение должно быть кроссплатформенным (Android и IOS версии).
- Мобильное приложение должно присылать push-уведомление с напоминанием о необходимости сделать заказ на следующий день.
- Веб-приложение должно поддерживаться всеми современными браузерами (Firefox, Chrome, Safari, Opera).
- Должна иметься возможность занести блюдо в чёрный список, если оно не устраивает пользователя.
- После того, как был сделан заказ, система должна формировать csv файл по шаблону, определенному службой доставки «Александра» и высылаться на их почтовый адрес. В этом файле должна содержаться информация о заказах всех сотрудников на следующий день.

На основе этих требований были поставлены следующие задачи, после выполнения которых web-сервис будет считаться минимально жизнеспособным продуктом (все нижепоставленные задачи актуальны для веб-версии и для мобильного приложения):

- реализовать функционал авторизации сотрудника;
- реализовать получение данных от сервера об актуальном меню;

- разработать модуль корзины, который должен отслеживать дневной лимит в 200 рублей на сотрудника;
- разработать алгоритм добавления блюда в «Чёрный список»;
- организовать хранение данных о заказах сотрудников в БД, а также формирование csv файла, содержащего таблицу с заказами всех сотрудников и отправкой этого файла на почту службы доставки «Александра»;
- разработать и реализовать единый дизайн для мобильного приложения и для веб-клиента;
- тестирование веб-клиента и мобильного приложения автоматизированными тестами и командой тестировщиков в компании;
- публикация мобильного приложения в PlayMarket и AppStore.

## **2.2 Выбор программных средств, технологии программирования для серверной части**

Для разработки серверной части был выбран язык PHP, и соответственно PHP-фреймворк Laravel.

Laravel – PHP-фреймворк с открытым исходным кодом на основе MVC, используемый для веб-разработки. Он основан на платформе Symfony - наборе повторно используемых компонентов и библиотек - что облегчает создание высокопроизводительных приложений.

Он предоставляет встроенные компоненты для часто используемых задач веб-приложений, таких как аутентификация, кэширование, сессии, что, возможно, является его самым большим преимуществом.

Преимущества Laravel:

- Хранение – отдельное место в структуре проекта.
- Eloquent - ORM для работы с базой данных, реализация шаблона ActiveRecord. В дополнение к стандартным операциям CRUD, также имеет: методы доступа и мутаторы, безопасное удаление, области запросов, отношения, модель наблюдателей.

- Artisan - позволяет управлять командной строкой.
- Простая маршрутизация, проверка входящих параметров.
- Интегрированный SCSS и сборщик скриптов.
- Встроенный шаблонный движок Blade.
- Обеспечивает эффективный API для быстрой отправки электронной почты (пригодится при отправке заказа по Email).
- Очереди заданий, планировщик заданий.
- Имеет отличные системы миграции баз данных.
- Имеет интегрированную систему модульного тестирования.
- Поддержка веб-сокетов для создания интерактивных приложений.
- Плагины ACL.
- Частые выпуски новых версий.

Из недостатков можно выделить отсутствие встроенных шаблонов интерфейса.

Во время разработки рекомендуется покрывать каждый из доступных маршрутов тестами, заполнять базу данных проверочными данными, придерживаться основы философии Laravel – краткости [6].

Laravel дает разработчику большую свободу в выборе решений и способах их реализации. Это расширяет круг возможностей для разработчика, но и накладывает на него определенную ответственность за успешность выбранного подхода.

### **2.3 Выбор программных средств, технологии программирования для веб-клиента**

При выборе фреймворка для разработки веб-клиента был сделан акцент на использование компонентного подхода, который предоставляют фреймворки React и Vue.

React и Vue очень похожи в своем подходе к работе с DOM (объектная модель документа). Они оба используют подход Virtual DOM для отрисовки

элементов и повторного рендеринга элементов в браузере. Фреймворки, использующие этот подход, хранят виртуальную копию DOM браузера. Затем они используют эту копию, чтобы определить, как лучше всего отобразить новые изменения.

Хотя React и Vue используют один и тот же подход к DOM, способ, которым они отображают HTML и CSS, различен.

Необходимо отметить, как React обрабатывает элементы рендеринга на веб-странице. React делает это с помощью компонентов. Базовый компонент React представлен в приложении Б.

Подход React к обработке HTML и CSS основан на использовании JSX. JSX позволяет разработчикам определять свои HTML-шаблоны (и часто CSS-рендеринг) в файлах Javascript. Обработка CSS может выполняться различными библиотеками (например, StyledComponents) [3].

В примере базового компонента React, представленного в приложении Б, использовался ReactDOM класс для поиска точки входа (в нашем случае div с идентификатором «root») и визуализации компонента App.

Также имеется возможность добавить несколько компонентов в приложение React, для этого нужно внедрить их в шаблон компонента. Данный пример также представлен в приложении Б.

Vue также использует компонентный подход для рендеринга HTML и CSS кода в браузере. Это означает, что фреймворк немного отличается от React. Подход Vue заключается в использовании HTML-шаблонов для определения способа визуализации компонентов.

Пример для компонента Vue представлен в приложении Б.

Подход Vue к визуализации HTML и CSS больше полагается на фактические шаблоны HTML, чтобы определить, как наши компоненты структурированы и упорядочены [4].

Это не исключает полностью шаблоны из файла компонента Vue. Разделы HTML всё ещё должны определяться в нашем компоненте. Тем не менее, именно в этом примере весь код обрабатывается Javascript.

Vue также позволяет использовать концепцию Single File Components. Этот подход позволяет включать CSS-стили, шаблоны и логику компонентов в один файл. Компоненты с одним файлом предлагают эффективные средства для включения CSS без необходимости установки специальной библиотеки поверх Vue. Однако, если у вас есть библиотека CSS, которую вы хотели бы использовать, Vue, скорее всего, ее поддержит. Такие библиотеки, как Vue Loader, помогают разработчикам использовать модули CSS в своих компонентах Vue.

Несмотря на то, что Vue имеет HTML-шаблонизацию из коробки, он также поддерживает JSX-шаблоны через расширение Babel. То, как Vue работает с помощью JSX, отличается от React, но предоставляет разработчикам возможность использовать Vue, не отказываясь от JSX.

## **2.4 Выбор программных средств, технологии программирования для мобильного приложения**

При выборе фреймворка для разработки мобильного приложения был сделан акцент на использование компонентного подхода, который предоставляют фреймворки React Native и Vue Native.

Что такое React Native? Фреймворк для создания собственных мобильных приложений с помощью React. React Native позволяет создавать приложения мирового класса на собственных платформах, используя согласованные возможности разработчиков на основе JavaScript и React. В центре внимания React Native - эффективность работы разработчиков на всех платформах, которые вам интересны. Facebook использует React Native в нескольких производственных приложениях и будет продолжать инвестировать в React Native [2].

Что такое Vue Native? Построение кроссплатформенных мобильных приложений с использованием Vue.js. Vue Native – это мобильная платформа для создания действительно нативного мобильного приложения с использованием Vue.js. Vue Native – это оболочка для API-интерфейсов React

Native, позволяющая использовать Vue.js и создавать многофункциональный мобильный пользовательский интерфейс [5].

Для разработки мобильного приложения выбор был сделан в пользу react native, ввиду того, что данный фреймворк обладает активным сообществом, что позволяет быстрее решать возникающие вопросы в ходе разработки. Также в компании ООО «Умный мир» есть несколько других проектов, где react native доказал свою состоятельность.

На основе анализа документации программного продукта было выявлено, что существует два направления запуска разрабатываемого приложения: React native init(CLI) или Expo. Сравнительный анализ этих способов приведён в таблице 1.

Таблица 1 – Сравнительный анализ React native init и Expo

Наименование	React native init(CLI)	Expo
Добавление модулей Java/Objective-C	Да	Нет
Вес стандартного приложения “Hello World”	5мб	25мб
Требуется Android Studio и XCode для запуска проектов	Да	Нет
Шрифты необходимо импортировать вручную в XCode	Да	Нет
Поделиться приложением через .apk	Да	Нет(QR-code)

Возможность открыть проект во время работы над ним	Нет	Да
--	-----	----

Ввиду меньшего веса стандартного приложения, а также поддержки возможности поделиться приложением через .apk, выбор был сделан в пользу react native init(CLI).

## **2.5 Управление глобальным состоянием в клиентских приложениях**

В пунктах 2.3 и 2.4 данной выпускной квалификационной работы было сказано, что в клиентских приложениях React JS и Vue JS используется компонентный подход. В связи с этим, у каждого компонента имеется своё состояние, в которое можно помещать необходимые данные. Однако, в крупных проектах возникает проблема, если данные какого-то из компонентов также нужны и в других компонентах. Таким образом, для решения указанных проблем следует применять Flux-архитектуру.

Основная идея довольно простая – создаётся одно хранилище для всех компонентов, чтобы каждый из компонентов имел возможность работать с данными хранилища: добавлять, изменять, удалять.

Для этого у каждого из фреймворков существуют свои библиотеки: у React – Redux, у Vue имеется собственное решение Vuex.

Принцип работы следующий: каждый компонент может через функцию, называемую action, отправить запрос на изменение данных в хранилище, вызывая для этого определенного типа reducer. Reducer, приняв изменяемые или новые данные от action, делает указанные действия с хранилищем, может также осуществлять предварительную обработку данных. Глобальное хранилище обновляется, и всем остальным компонентам, которые работают с этими данными, также доступны актуальные данные из хранилища. Схематично данная архитектура представлена на рисунке 2.



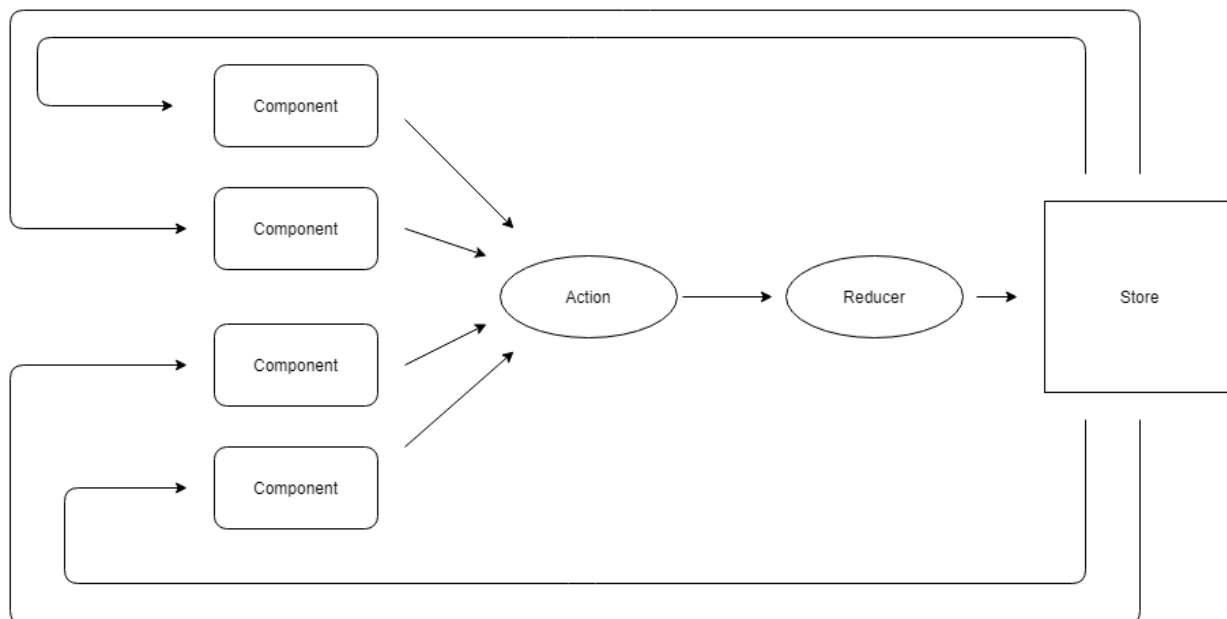


Рисунок 2 – Принцип работы с Flux-архитектурой

Так как и в мобильном приложении, и в веб-приложении используется компонентный подход, было также решено использовать глобальное хранилище для хранения данных, приходящих от сервера.

## 2.6 Работа с Git

В больших проектах с многочисленной командой разработчиков необходима система контроля версий, которая позволит нескольким разработчикам работать параллельно над одним проектом, сохраненным в удаленном хранилище. Удалённое хранилище также нужно для сохранности данных в случае их потери на физическом носителе. В проекте SmartFood в качестве удалённого хранилища использовался репозиторий Gitlab.

Преимущества Git над другими системами контроля версий в том, что большинство операций доступны локально, без подключения к сети. Это даёт возможность работать над проектом в файловой системе без подключения к сети, а при возможности подключения к сети можно легко интегрироваться с удалённым репозиторием.

У Git есть три состояния, в котором могут находиться файлы:

- зафиксированное (committed) – означает, что файл уже сохранён в локальной базе;
- изменённое (modified) – означает, что файл претерпел изменения, которые ещё не были зафиксированы;
- подготовленное (staged) – измененные файлы, отмеченные для включения в следующий коммит.

Следовательно, у проекта Git есть три основные секции:

- Git-директория – это то место, где Git хранит метаданные и базу объектов вашего проекта. Это самая важная часть Git, и это та часть, которая копируется при клонировании репозитория с другого компьютера.
- Рабочая директория является «снимком» версии проекта. Файлы распаковываются из сжатой базы данных в Git-директории и располагаются на диске, для того чтобы их можно было изменять и использовать.
- Область подготовленных файлов – это файл, обычно располагающийся в вашей Git-директории, в нём содержится информация о том, какие изменения попадут в следующий коммит [1].

Схематично цикл изменения файлов представлен на рисунке 3.

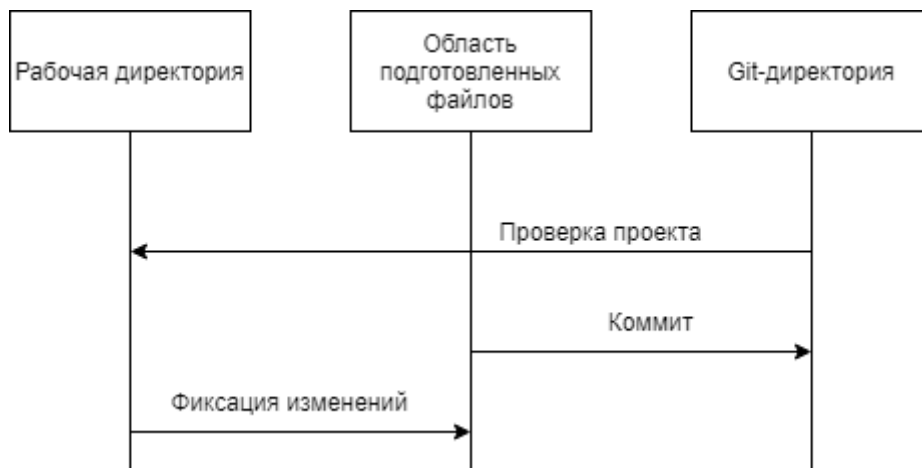


Рисунок 3 – цикл изменения файлов в Git

## 2.7 Организация хранения данных

Хранение данных осуществляется с помощью PostgreSQL Server. PostgreSQL – это СУБД, предоставляющая реализацию языка запросов SQL.

При создании таблиц базы данных была учтена тематика web-сервиса кейтеринговых услуг.

При работе с проектом все пользователи были разделены на 2 основные группы, такие как зарегистрированные пользователи и незарегистрированные пользователи. Полный функционал доступен только зарегистрированному пользователю, для незарегистрированного доступен только функционал авторизации.

Для хранения информации о блюдах используется таблица Dish. В данной таблице хранится название блюда, описание блюда и вес блюда (при наличии), и цена. Для соблюдения 1 НФ категория блюда вынесена в отдельную таблицу Dish\_Category. Таблица Employee хранит сведения о сотрудниках, такие как ФИО, день рождения и работает ли сотрудник в компании в данный момент. Также есть таблицы Menu и Menu\_Dish, предназначенные для сортировки блюд по дням. Сформированные заказы попадают в таблицу Employee\_Orders. Чтобы упростить взаимодействие пользователя с корзиной, было решено вывести отдельную таблицу Basket, в которой будет храниться информация о товарах, добавленных в корзину. Схема спроектированной базы данных изображена на рисунке 4.

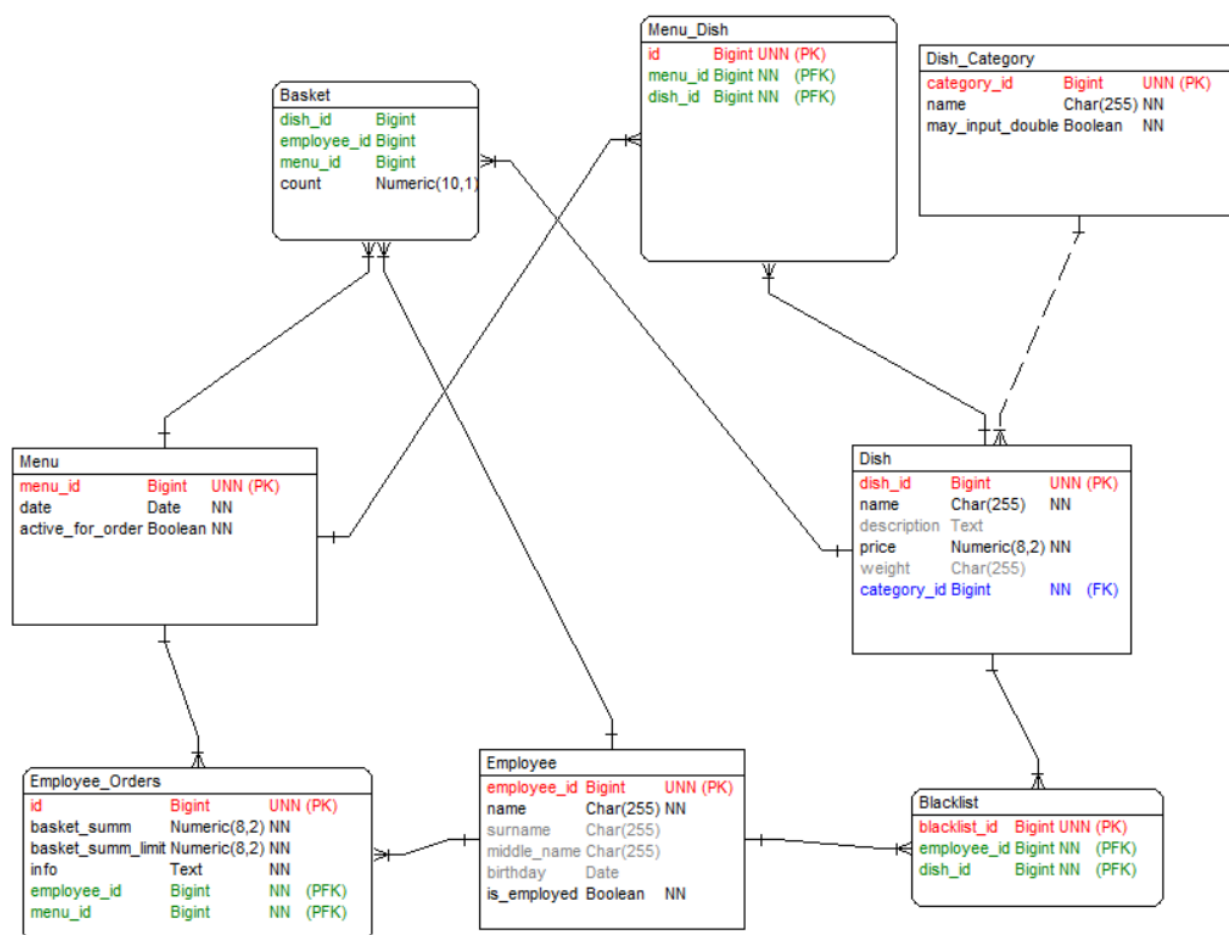


Рисунок 4 – физическая схема БД

Данная схема реляционной базы данных нормализована, т.е. соблюдены три нормальные формы. Это позволяет обеспечить минимальную логическую избыточность [10].

## 2.8 Диаграмма вариантов использования web-сервиса

С полным функционалом приложения может работать только авторизованный пользователь. Неавторизованному пользователю доступна только страница входа в приложение, где он должен авторизоваться через свою корпоративную почту.

Авторизованный пользователь может ознакомиться с текущим меню, добавить блюдо в корзину или удалить его оттуда, изменить количество порций для блюда в корзине, посмотреть свой лимит, который нельзя

превысить, посмотреть сумму заказа в корзине и на странице меню, отправить заказ.

Также для авторизованного пользователя имеется функционал добавления блюда в чёрный список, если пользователь не хочет по каким-то причинам видеть это блюдо в своём меню.

Диаграмма вариантов использования изображена на рисунке 5.

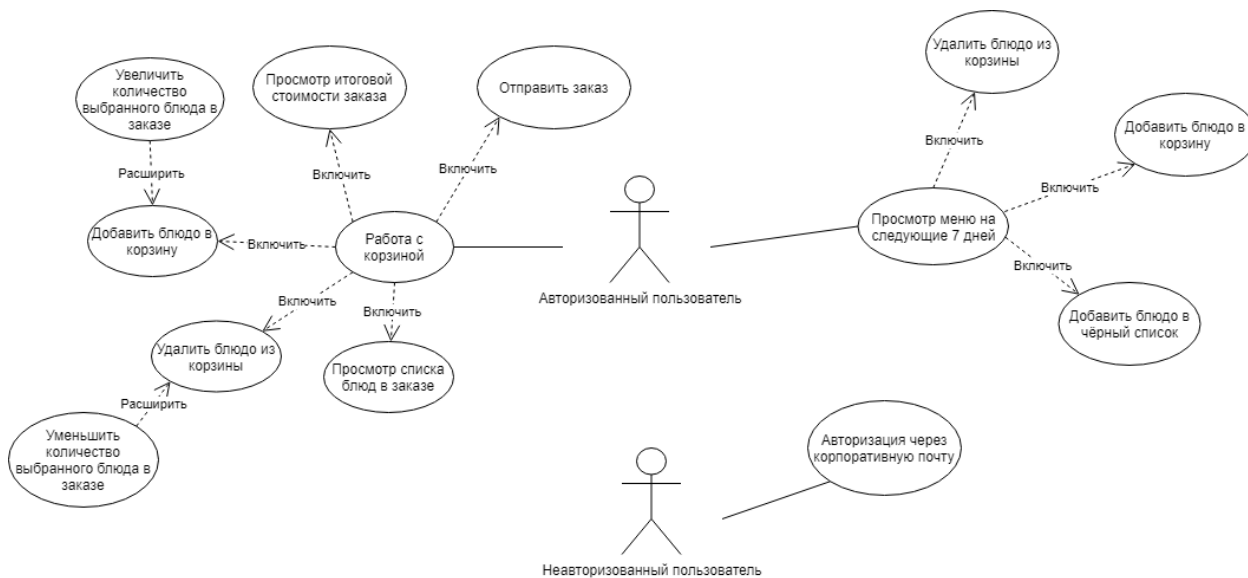


Рисунок 5 – диаграмма вариантов использования web-сервиса

## 2.9 Диаграмма последовательности

На рисунке 6 представлена диаграмма последовательности для процесса авторизации пользователя.

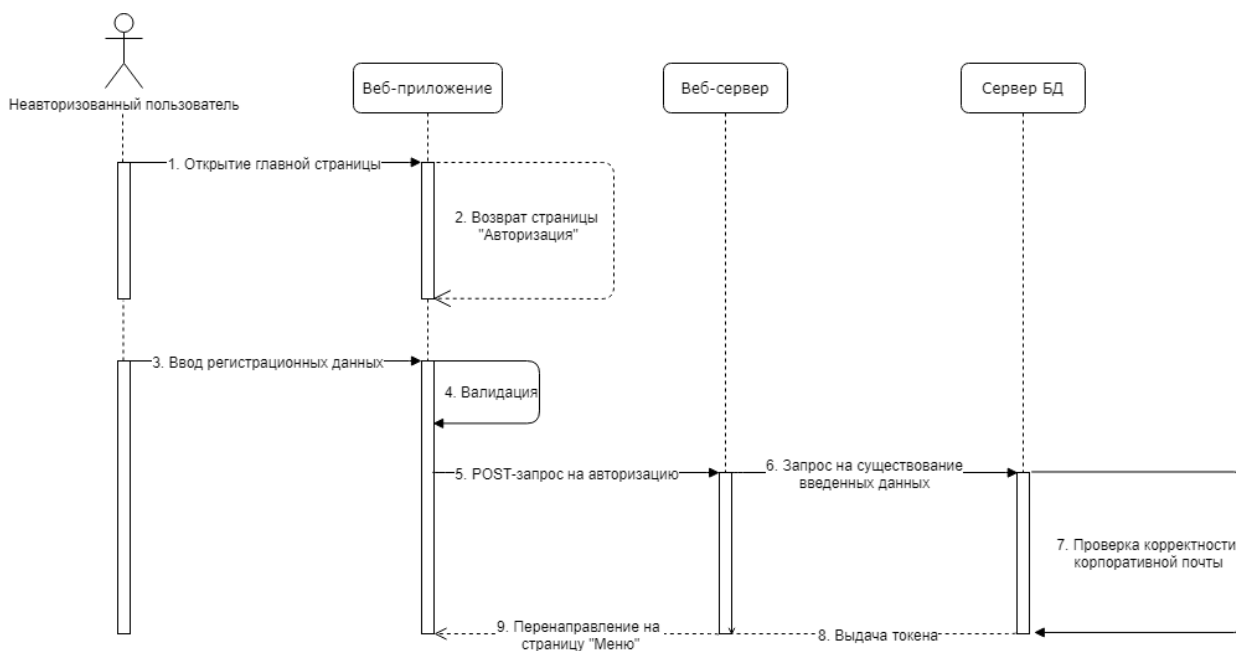


Рисунок 6 – диаграмма последовательности процесса авторизации пользователя

Процесс авторизации довольно стандартный. Пользователь вводит необходимые данные, сервер проверяет введенные данные, и, если они верны, выдаёт токен, с которым уже у веб-клиента или мобильного приложения имеется доступ к меню, а также рефреш-токен, чтобы была возможность обновить токен, который истёк, и пользователь может приступить к оформлению заказа.

### 2.10 Диаграмма деятельности «Оформление заказа»

Выбор блюд и оформление заказа в web-сервисе сделано по определенному алгоритму, представленному ниже.

1. Пользователь, предварительно авторизовавшись, переходит на страницу «Меню» за текущий день, где выбирает понравившиеся ему блюда (может указать количество порций). В мобильном приложении кнопки отображения действий над блюдом спрятаны за элементом списка, и появляются при действии смахивания позиции влево или вправо. Из доступных действий у пользователя: добавить блюдо в чёрный список, добавить в корзину, удалить из корзины.

2. Затем пользователь переходит в корзину, где видит подробную информацию о своём заказе: перечень позиций, количество порций каждого из блюд и общую стоимость заказа. Если общая стоимость заказа превышает 200 рублей, кнопка оформления заказа становится недоступной. Пользователю нужно будет отредактировать свой заказ, чтобы подкорректировать стоимость заказа.

3. Пользователь нажимает кнопку оформить заказ, и видит сообщение об успешном создании заказа или сообщение об ошибке.

Данный алгоритм также представлен с помощью диаграммы деятельности на рисунке 7.

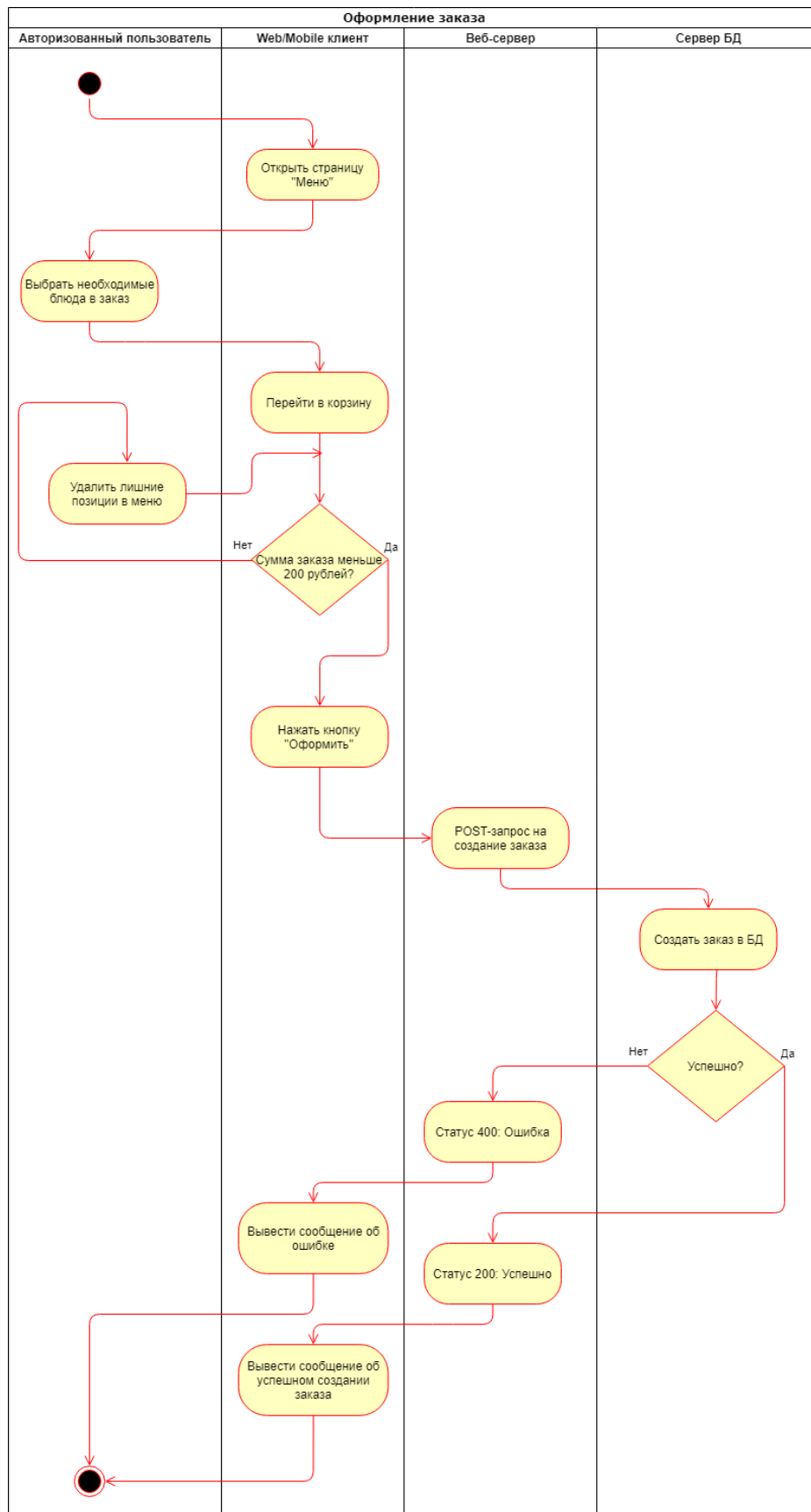


Рисунок 7 – диаграмма деятельности «Оформление заказа»



## 2.11 Раздел авторизации

Механизм авторизации был выбран довольно простой. У каждого сотрудника компании имеется корпоративная почта, вводя которую, он авторизуется в приложении.

Было принято решение отказаться от пароля, т.к. нет необходимости защищать какие-то секретные данные пользователя. В профиле не указаны персональные данные, а также это удобно тем, что, если один из сотрудников забыл заказать себе обед, он может попросить другого сотрудника, чтобы тот сделал за него заказ, и не нужно будет для этого сообщать пароль. Корпоративные почты сотрудников хранятся в документе, по которому можно найти этого сотрудника и авторизоваться за него.

Пример страницы авторизации представлен на рисунке 8.

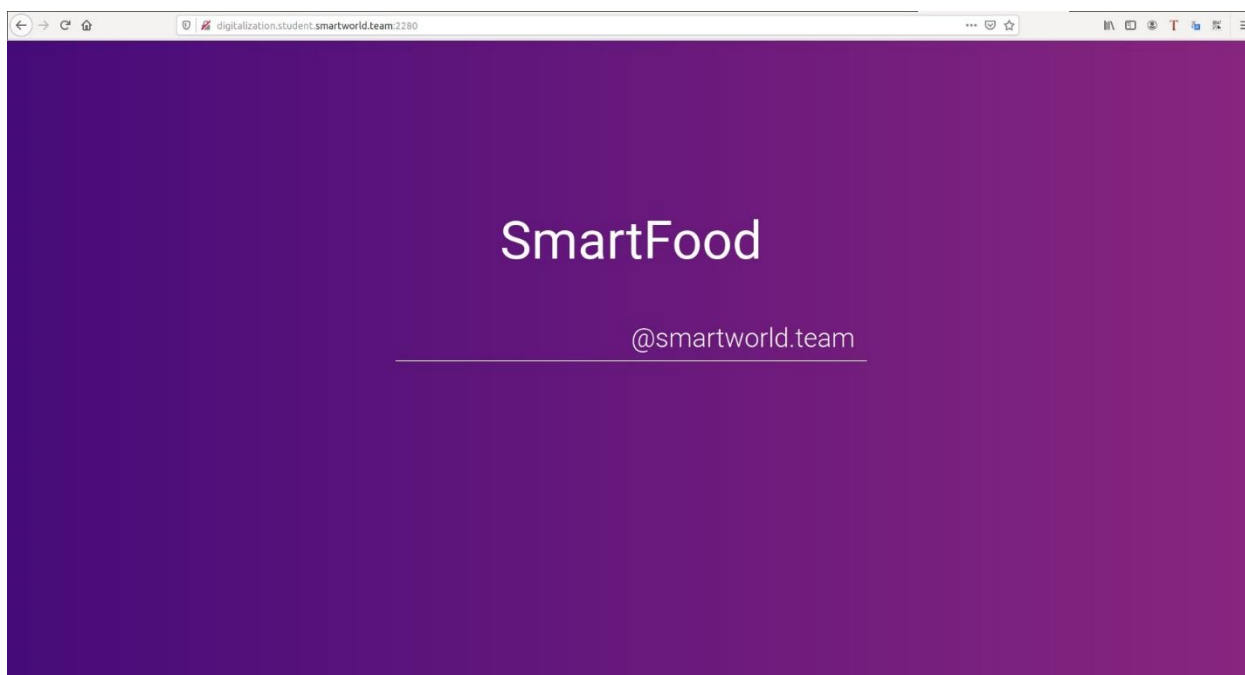


Рисунок 8 – Пример страницы авторизации в веб-приложении

Пример страницы авторизации в мобильном приложении представлен на рисунке 9.

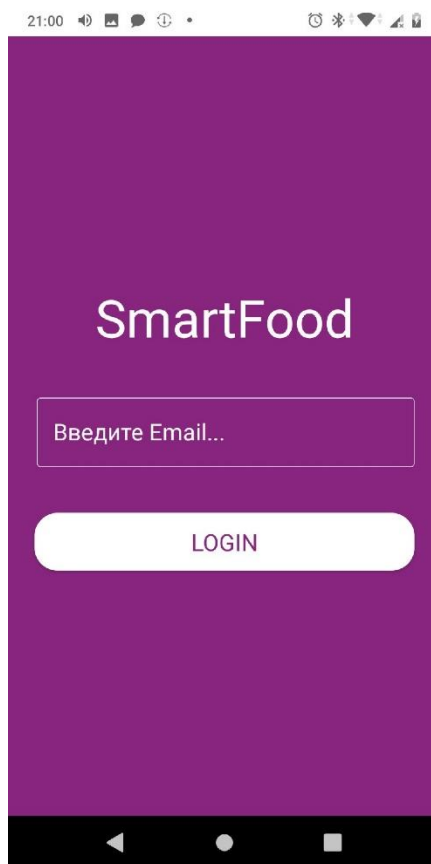


Рисунок 9 – Пример страницы авторизации в мобильном приложении

## 2.12 Раздел «Меню»

Раздел меню должен показывать пользователю отсортированные по категориям блюда (первые блюда, вторые блюда и т.п.). У каждого блюда показывается цена, также если оно выбрано в корзину, имеется возможность увеличить или уменьшить количество порций. В правом верхнем углу есть возможность перейти в корзину, также указана текущая сумма набранных продуктов в корзине. При превышении лимита в 200 рублей сумма становится красной, что говорит пользователю, что он превысил лимит.

Для каждого блюда также имеется определенный функционал:

- Добавить в чёрный список – блюдо при желании может быть добавлено в чёрный список, в этом случае оно не отображается для пользователя. Но в настройках профиля можно выбрать и такой вариант, что блюдо будет отображаться, но на сером фоне.

- Добавить в корзину – пользователь может добавить блюдо в корзину нажатием на блок блюда.
- Увеличить/уменьшить количество порций в корзине – становится доступным после того, как блюдо оказалось в корзине. Не нужно переходить в корзину, чтобы отредактировать количество выбранных блюд.

Пример отображения раздела «Меню» в веб-приложении представлен на рисунке 10.

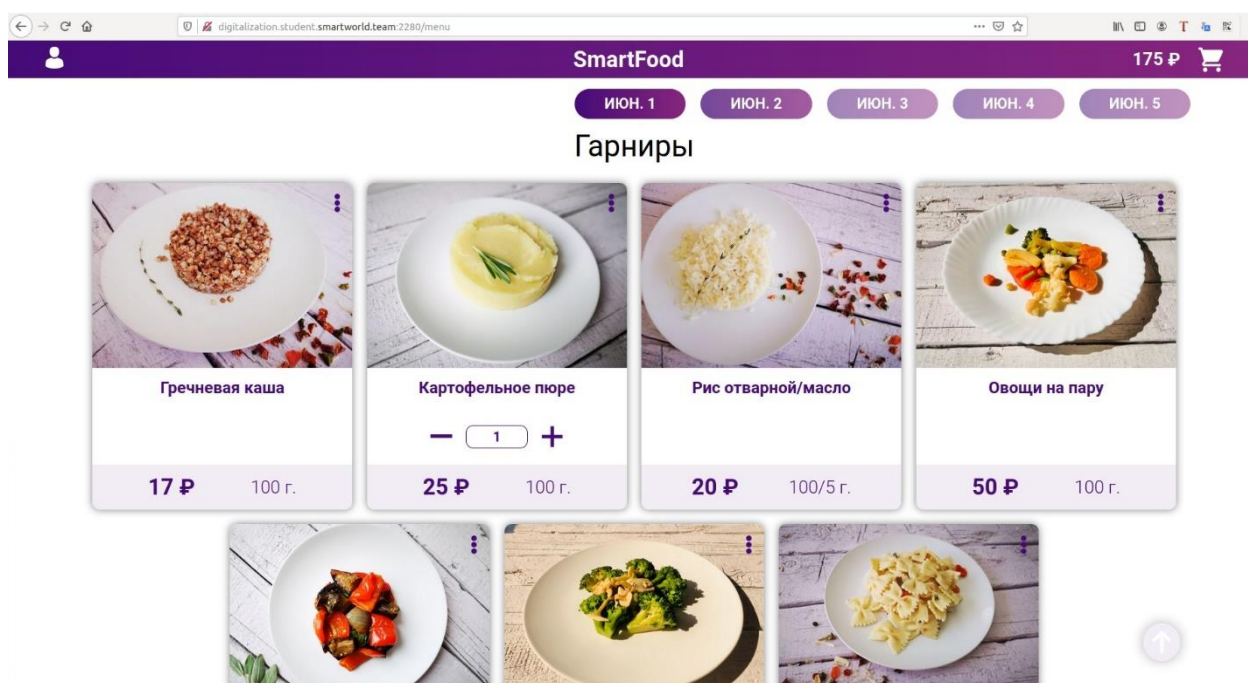


Рисунок 10 – Страница «Меню» в веб-приложении

В мобильном приложении меню представлено списком, элементы которого можно сдвигать влево или вправо. При сдвиге элемента меню отображаются кнопки действий: добавить в корзину, удалить из корзины, добавить в чёрный список.

Пример отображения раздела «Меню» в мобильном приложении представлен на рисунке 11.



Рисунок 11 – Страница «Меню» в мобильном приложении

Как было сказано ранее, при добавлении блюда в чёрный список оно или перестаёт отображаться, или отображается на сером фоне при включенном «Черном списке». Пример отображения блюда из чёрного списка представлен на рисунке 12.

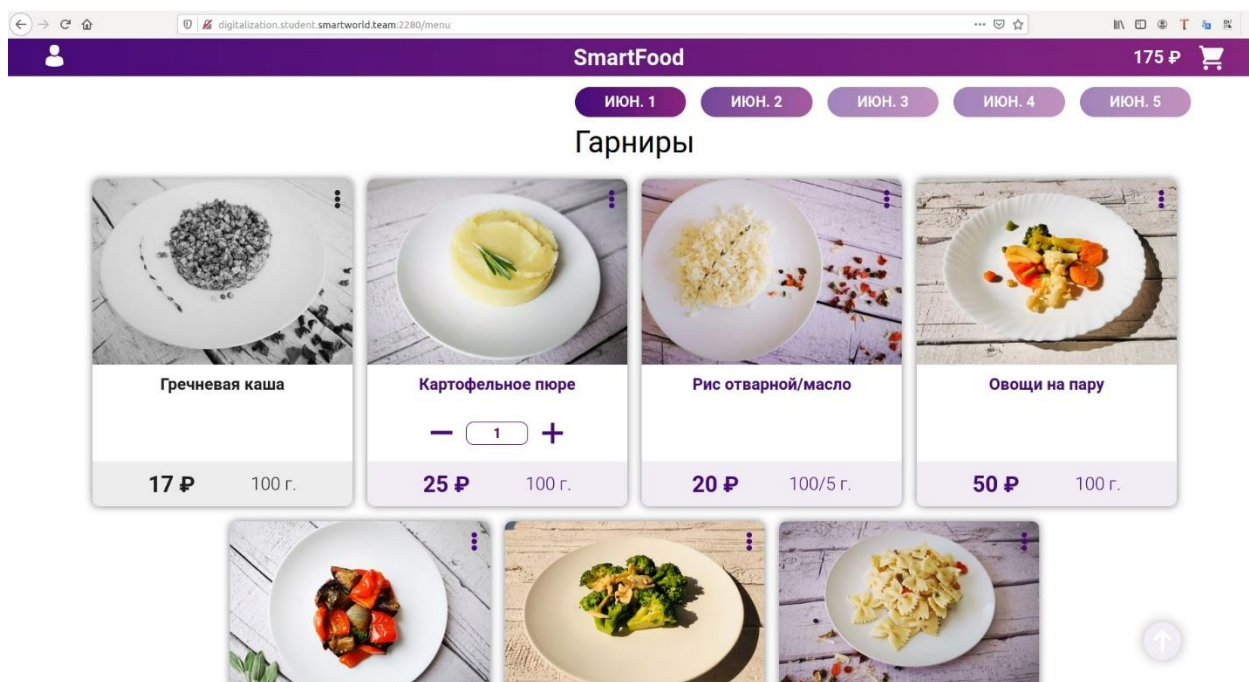


Рисунок 12 – Отображения блюда «Гречневая каша» в чёрном списке

Laravel предоставляет единое API для различных сервисов очередей, таких как Beanstalk, Amazon SQS, Redis или даже реляционных БД. Очереди позволяют отложить выполнение времязатратных задач, таких как отправка e-mail, на более позднее время, таким образом на порядок ускоряя обработку веб-запросов в приложении. В данном случае, используется Redis, и очереди предназначены для обновления информации об актуальном меню каждые 4 часа.

Также имеется настройка кэширования прямо в Redis, что позволяет серверу доставать актуальные данные не из базы данных, а из кэша Redis. Таким образом, данный способ уменьшает время работы запроса к данным.

### 2.13 Раздел «Профиль»

В разделе «Профиль» в модальном окне пользователю отображается его имя и почта, а также доступна опция отключения/включения отображения блюд из чёрного списка. Пример раздела «Профиль» в веб-приложении представлен на рисунке 13.

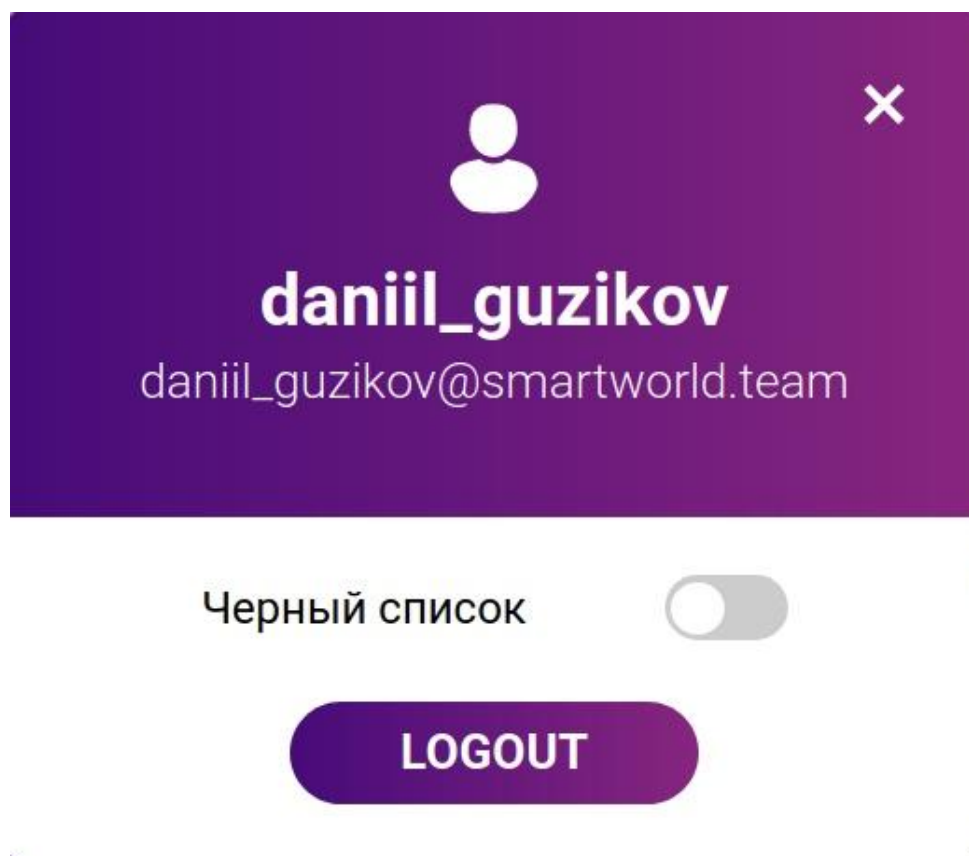


Рисунок 13 – раздел «Профиль» в веб-приложении

Пример раздела «Профиль» в мобильном приложении представлен на рисунке 14.

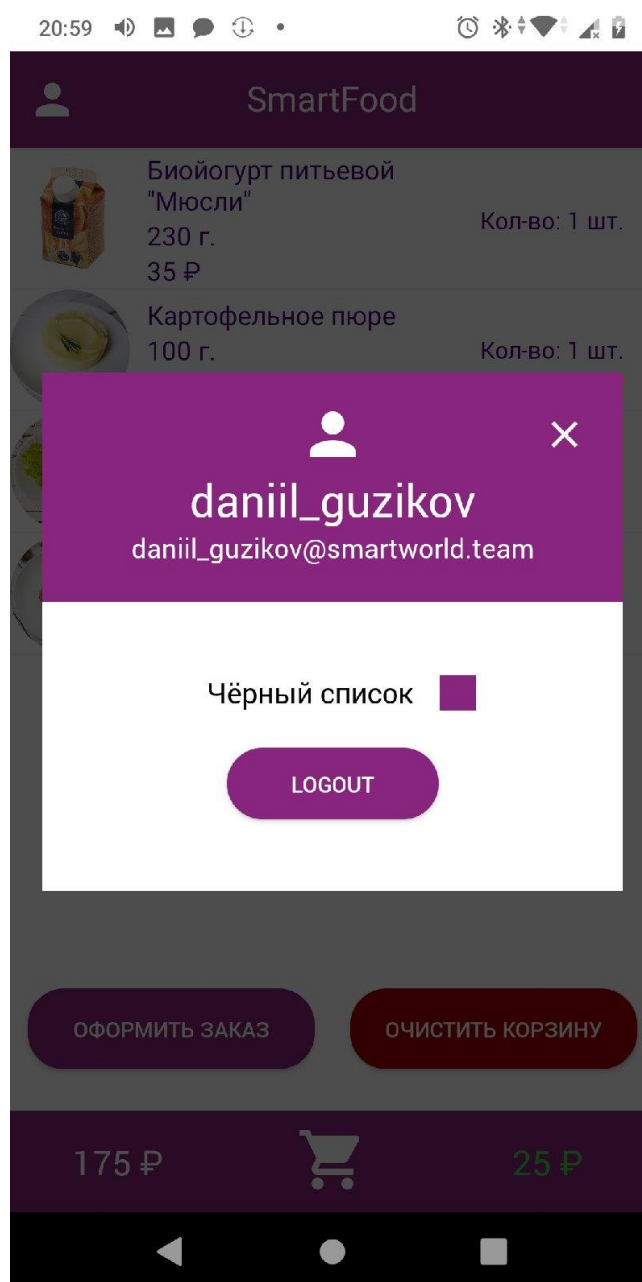


Рисунок 14 – раздел «Профиль» в мобильном приложении

## 2.14 Раздел «Корзина»

Корзина открывается в модальном окне. Здесь стандартный функционал для подобных модулей: увеличение/уменьшение количества порций у каждого блюда, отображение стоимости каждого блюда, а также веса, если он есть. Имеется возможность очистить всю корзину или отправить заказ. Пример отображения корзины представлен на рисунке.



Здесь также отслеживается лимит в 200 рублей. При превышении лимита, кнопка «Оформить заказ» становится недоступной.

Пример отображения модуля «Корзина» в веб-приложении представлен на рисунке 15.

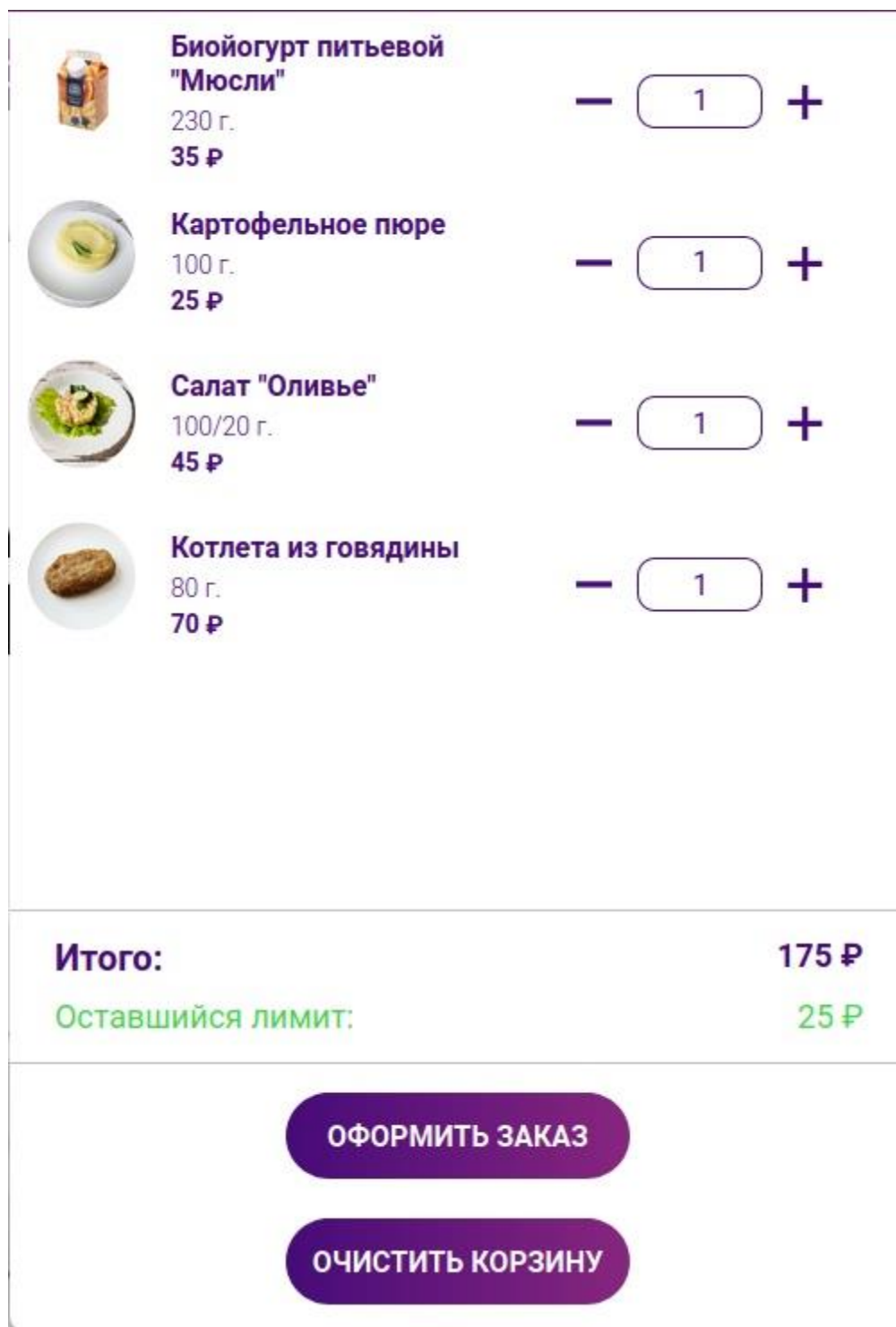


Рисунок 15 – Отображение модуля «Корзина» в веб-приложении

Пример отображения модуля «Корзина» в мобильном приложении представлен на рисунке 16.



Рисунок 16 - Отображение модуля «Корзина» в мобильном приложении



### **3. Финансовый менеджмент, ресурсоэффективность и ресурсосбережения**

#### **3.1 Оценка коммерческого потенциала и перспективности проведения научных исследований с позиции ресурсоэффективности и ресурсосбережения**

Целью представленной магистерской диссертации является проектирование и программная реализация web-сервиса кейтеринговых услуг для компании ООО "Умный мир". Разрабатываемый сервис предназначен для заказа еды в службе доставки "Александра" сотрудниками компании ООО "Умный мир". Разработано веб-приложение, а также мобильное приложение, написанное с помощью react-native – фреймворк для разработки кроссплатформенных приложений для IOS и Android.

Для успешного достижения цели необходимо выполнить ряд задач и дать оценку коммерческих возможностей проведенного исследования, выявить его ресурсосберегающий потенциал, определить финансовую эффективность исследования.

В данной главе дипломного проекта оценивается экономический эффект от применения разработанных и модернизированных систем проекта, а также ресурсоэффективность проекта.

Целью раздела является определение перспективности научно-исследовательского проекта. Задачами раздела являются:

- оценка коммерческого потенциала и перспективности проведения научных исследований с позиции ресурсоэффективности и ресурсосбережения;
- планирование работ по научно-исследовательскому проекту с использованием линейного графика;
- расчет бюджета научного-технического исследования;

- определение экономической эффективности исследования.

### 3.1.1 Потенциальные потребители результатов исследования

Потенциальными потребителями являются работники компании ООО «Умный мир», которые с помощью данной системы имеют возможность заказать обед в офис с минимальными тратами личных ресурсов и ресурсов компании. Также, в будущем, web-сервис может быть реализован в другие компании, которые захотят выстроить процесс данным образом.

### 3.1.2 Анализ конкурентных технических решений

Анализ аналоговых решений позволяет выявить недостатки своего продукта и повысить, тем самым, его конкурентоспособность. Так же анализ учитывает достоинства своего продукта и недостатки продуктов конкурентных.

Но в нашем случае, web-сервис, разработанный для компании «Умный мир», является абсолютно оригинальным и не имеет аналогов, поскольку разработан специально для компании.

### 3.1.3 Определение возможных альтернатив проведения научных исследований

С помощью морфологического подхода можно определить возможные альтернативы в проведении научных исследований. Рассмотрим варианты разработки для мобильного приложения. Данный подход и его результат использования представлены в таблице 2.

Таблица 2 – Морфологическая матрица

	1	2	3	4
А: Язык программирования	Java	C#	JavaScript	Dart
Б: Среда разработки	Android Studio	Visual studio	VS Code	Flutter

В: Android	Эмулятор Android SDK emulator	Genymotion	Android SDK emulator	Android SDK emulator
---------------	--	------------	----------------------------	----------------------------

Исходя из составленной морфологической матрицы, можно получить как минимум 4 варианта реализации и направления научных исследований при работе над проектом:

- Исполнение 1. А1Б1В1
- Исполнение 2. А2Б2В2
- Исполнение 3. А3Б3В3
- Исполнение 4. А4Б4В4

В качестве варианта реализации было выбрано «Исполнение 3». Данный вариант подразумевает использование языка Javascript и фреймворка react-native для разработки кроссплатформенных приложений. В качестве официальной среды разработки используется VS Code. Язык Javascript также хорош тем, что используется как и в веб-разработке, так и в мобильной разработке, что идеально подходит при разработке рассматриваемого web-сервиса кейтеринговых услуг.

«Исполнение 1» подразумевает использования языка Java, данный вариант реализации считается устаревшим.

«Исполнение 2» и «Исполнение 4» также являются кроссплатформенными вариантами реализациями, но обладают меньшим авторитетом, чем react-native. Помимо этого, среда разработки Flutter на данный момент не имеет удовлетворяющий функционал в сравнение с VS Code.

## 3.2 Планирование проектных работ

### 3.2.1 Структура работ в рамках проекта

Планирование комплекса предполагаемых работ осуществлено в следующем порядке:

- определение структуры работ проекта;
- определение участников каждой работы;
- установление продолжительности работ;
- построение графика проведения проектной работы.

В данном разделе составлен перечень этапов и работ проекта, а также произведено распределение исполнителей по видам работ, таблица 3.

Таблица 3 – Перечень этапов и работ проекта

Основные этапы	№ раб	Содержание работ	Исполнители и их загрузка(занятость)
Обсуждение идеи технологии	1	Аналитика и обсуждения	Аналитик-100%, Разработчик-100%, Дизайнер-100%, Тестировщик-100%, Научный руководитель-100%
Проектирование	2	Разработка единого дизайна web-сервиса	Дизайнер-100%
	3	Анализ и обзор технологий	Разработчик-100%
	4	Разработка архитектуры приложения	Разработчик-100%

Разработка web-сервиса	5	Реализация функционала приложения	Разработчик-100%
	6	Тестирование	Разработчик-43%, Тестировщик-57%
Подготовка ВКР	7	Составление отчёта	Научный руководитель – 60% Разработчик – 100%

### 3.2.2 Определение трудоемкости выполнения работ

Трудовые затраты в большинстве случаев образуют основную часть стоимости разработки, поэтому важным моментом является определение трудоемкости работ каждого из участников проекта.

Трудоемкость выполнения проекта оценивается экспертным путем в человеко-днях и носит вероятностный характер, т.к. зависит от множества трудно учитываемых факторов. Для определения, ожидаемого (среднего) значения трудоемкости  $t_{ож\ i}$  используется следующая формула:

$$t_{ож\ i} = \frac{3t_{min\ i} + 2t_{max\ i}}{5} \quad (1),$$

$t_{ож\ i}$  - минимально возможная трудоемкость выполнения заданной  $i$ -ой работы (оптимистическая оценка: в предположении наиболее благоприятного стечения обстоятельств), чел.-дн.:  $t_{max\ i}$  – максимально возможная трудоемкость выполнения заданной  $i$ -ой работы (пессимистическая оценка: в предположении наиболее неблагоприятного стечения обстоятельств), чел.-дн. Расчеты  $t_{ож\ i}$  занесены в таблицу 4.

Таблица 4 – Временные показатели научного исследования

Этап	Исполнители	Продолжительность работ, дни			Трудоемкость работ по исполнителям чел.- дн.									
					Т <sub>рд</sub>					Т <sub>кд</sub>				
		t <sub>min</sub>	t <sub>max</sub>	t <sub>ож</sub>	Аналитик	Дизайнер	Разработчик	Тестировщик	Научный руководитель	Аналитик	Дизайнер	Разработчик	Тестировщик	Научный руководитель
1	2	3	4	5	6	7	8	9	10	11	12	13		
Аналитика и обсуждения	Аналитик, дизайнер, разработчик, тестировщик, научный руководитель	1	2	1,4	0,46	0,46	0,46	0,46	0,46	0,5543	0,5543	0,5543	0,5543	0,5543
Разработка единого дизайна	Дизайнер	4	8	5,6	-	5,6	-	-	-	-	6,748	-	-	-
Анализ и обзор технологий	Разработчик	3	9	5,4	-	-	5,4	-	-	-	-	6,507	-	-
Разработка архитектуры приложения	Разработчик	2	3	2	-	-	2	-	-	-	-	2,41	-	-
Реализация функционала приложения	Разработчик	25	47	37	-	-	37	-	-	-	-	44,585	-	-
Тестирование	Разработчик, тестировщик	7	14	7	-	-	3	4	-	-	-	3,615	4,82	-
Оформление ВКР	Разработчик, научный руководитель	28	28	28	-	-	28	-	28	-	-	33,74	-	33,74
<b>Итого:</b>				<b>86,4</b>	<b>0,46</b>	<b>6,06</b>	<b>75,86</b>	<b>4,46</b>	<b>28,46</b>	<b>0,5543</b>	<b>7,3023</b>	<b>91,4113</b>	<b>5,3743</b>	<b>34,2943</b>

Исходя из ожидаемой трудоемкости работ, определяется продолжительность каждой работы в рабочих днях  $T_p$ , учитывающая параллельность выполнения работ несколькими исполнителями. Такое вычисление необходимо для обоснованного расчета заработной платы, так как удельный вес зарплаты в общей сметной стоимости проекта составляет около 65 %.

$$T_{pi} = \frac{t_{ожi}}{ч_i} \quad (2),$$

где  $T_{pi}$  – продолжительность одной работы, раб. дн.;

$t_{ожi}$  – ожидаемая трудоемкость выполнения одной работы, чел.-дн.

$ч_i$  – численность исполнителей, выполняющих одновременно одну работу.

### **3.2.3 Разработка графика проведение проекта**

На основе таблицы 4 построен календарный план-график для максимального по длительности исполнения работ в рамках выполняемого проекта (таблица 5)

Таблица 5 – линейный график работ

Этап	Аналитик	Научный руководитель	Дизайнер	Разработчик	Тестирующая	Март			Апрель			Май			Июнь	
						10	20	30	40	50	60	70	80	90	100	110
1	0,5		0,5	0,5	0,5											
2	-		6,7	-	-											
3				6,5												
4				2,4												
5				44,5												
6				8,4	8,4											
7		33,7		33,7												

Аналитик – Дизайнер – Разработчик –

Тестирующая – Научный руководитель –



### 3.2.4 Бюджет научно-технического исследования (НТИ)

При планировании бюджета НТИ должно быть обеспечено полное и достоверное отражение всех видов расходов, связанных с его выполнением. В процессе формирования бюджета НТИ используется следующая группировка затрат по статьям:

- материальные затраты НТИ
- основная заработная плата исполнителей темы
- дополнительная заработная плата исполнителей темы
- отчисления во внебюджетные фонды (страховые отчисления)
- накладные расходы

### 3.2.5 Расчет материальных затрат НТИ

Расчет материальных затрат осуществляется по следующей формуле:

$$Z_m = (1 + k_t) * \sum_{i=1}^m C_i * N_{расхi} \quad (3),$$

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

$N_{расхi}$  – количество материальных ресурсов  $i$ -го вида, планируемых к использованию при выполнении научного исследования (шт., кг, м,  $m^2$  и т.д.);

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

$k_t$  – коэффициент, учитывающий транспортно-заготовительные расходы. Значения цен на материальные ресурсы могут быть установлены по данным, размещенным на соответствующих сайтах в интернете предприятиями изготовителями (либо организациями-поставщиками).

Величина коэффициента ( $k_t$ ), отражающего соотношение затрат по доставке материальных ресурсов и цен на их приобретение, зависит от условий договоров поставки, видов материальных ресурсов, территориальной

удаленности поставщиков и т.д. Транспортные расходы принимаются в пределах 15-25% от стоимости материалов. Материальные затраты, необходимые для данной разработки, отсутствуют.

### 3.2.6 Основная заработная плата исполнителей темы

Основная заработная плата рассчитывается по формуле 4:

$$C_{\text{осн/зп}} = C_{\text{зп}i} * t_i * K_{\text{и}} \quad (4),$$

где  $C_{\text{зп}i}$  – среднедневная заработная плата работника, выполняющего  $i$ -ый вид работы, руб/день;

$t_i$  – затраты труда на выполнение  $i$ -го вида работ, в днях;

$K_{\text{и}}$  – интегральный коэффициент, для пятидневной рабочей недели  $K_{\text{и}}=1,59$ , а для шестидневной рабочей недели  $K_{\text{и}}=1,699$ .

Среднедневная заработная плата рассчитывается по формуле 5:

$$C_{\text{зп}i} = \frac{D}{N} \quad (5),$$

где  $D$  – месячный должностной оклад работника;

$N$  – количество рабочих дней в месяц: для пятидневной рабочей недели  $N=20,75$ , для шестидневной рабочей недели  $N=25,083$ ;

Расчет затрат на основную заработную плату приведен в таблице 6. Оклад разработчиков составляет 37000, оклад аналитика проекта составляет 30000, оклад дизайнера – 30000, оклад тестировщика – 25000, оклад научного руководителя – 33664.

Таблица 6 – Затраты на основную заработную плату

Исполнители	Оклад (руб./мес.)	Среднедневная заработная плата Сзп (руб./р.день)	Трудоемкость (ti), чел-дни	Коэффициент	Затраты на основную зарплату (руб.)
Аналитик	30000	1445,78	0,46	1,59	1057,44
Разработчики	37000	1783,13	75,86	1,59	215076,5
Дизайнер	30000	1445,78	6,06	1,59	13930,67
Тестировщик	25000	1204,82	4,46	1,59	8543,86
Научный руководитель	33664	1342,1	28,46	1,699	64895,29
<b>Итого</b>					<b>303503,76</b>

### 3.2.7 Расчет затрат на социальный налог

Затраты на единый социальный налог (ЕСН), включающий в себя отчисления в пенсионный фонд, на социальное и медицинское страхование, составляют 30 % от полной заработной платы по проекту. Затраты на единый социальный налог определяется исходя из формулы 6:

$$C_{\text{соц}} = C_{\text{осн/зп}} * 0,3 \quad (6),$$

Таким образом, затраты на единый социальный налог  $C_{\text{соц}} = 303503,76 * 0,3 = 91051,13$

### 3.2.8 Расчет затрат на электроэнергию

Данный вид расходов включает в себя затраты на электроэнергию, потраченную в ходе выполнения проекта на работу используемого оборудования, рассчитываемые по формуле:

$$C_{\text{эл.об.}} = P_{\text{об.}} * t_{\text{об.}} * Ц_{\text{э}} \quad (7),$$

где  $P_{\text{об.}}$  – мощность, потребляемая оборудованием, кВт;

$Ц_{\text{э}}$  – тариф на 1 кВт\*час;

$t_{\text{об.}}$  – время работы оборудования, час.

Для ТПУ  $Ц_{\text{э}} = 6,59$  руб./кВт\*час (с НДС)

Время работы оборудования ( $T_{\text{рд}}$ ) вычисляется на основе итоговых данных таблицы 5.2 для аналитика, дизайнера, тестировщика, разработчика в сумме из расчета, что продолжительность рабочего дня равна 8 часов.

$$t_{\text{об.}} = T_{\text{рд}} * K_t \quad (8),$$

где  $K_t$  – коэффициент использования оборудования по времени, равный отношению времени его работы в процессе выполнения проекта к  $T_{\text{рд}}$ , принят за 1(100%), так как работа перечисленных выше исполнителей это предполагает.

Мощность, потребляемая оборудованием, определяется по формуле:

$$P_{\text{об.}} = P_{\text{ном.}} * K_C \quad (9),$$

где  $P_{\text{ном.}}$  – номинальная мощность оборудования, кВт;

$K_C \leq 1$  – коэффициент загрузки, зависящий от средней степени использования номинальной мощности. Для технологического оборудования малой мощности  $K_C = 1$ .

Расчёт часов, потраченных на работу за компьютером:  $(0,46 + 6,06 + 75,86 + 4,46) * 8 = 86,84 * 8 = 694,72$

Таблица 7 – Затраты на электроэнергию

Наименование оборудования	Время работы оборудования $t_{\text{Об}}$ , час	Потребляемая мощность $P_{\text{Об}}$ , кВт	Затраты $\text{Э}_{\text{Об}}$ , руб.
Персональный компьютер	694,72*1	0,3	1373,46
<b>Итого:</b>			<b>1373,46</b>

### 3.2.9 Накладные расходы

Накладные расходы учитывают прочие затраты организации, не попавшие в предыдущие статьи расходов: печать и ксерокопирование материалов исследования, оплата услуг связи, электроэнергии и т.д. Их величина определяется по формуле 10:

$$Z_{\text{накл}} = (\text{сумма статей}) * k_{\text{нр}} \quad (10),$$

где  $k_{\text{нр}}$  – коэффициент, учитывающий накладные расходы.

Величина коэффициента накладных расходов равна 10%.

Исполнение 3:  $(303503,76+91051,13+1373,46) * 0,10 = 39592,84$

### 3.2.10 Формирование бюджета затрат научно-исследовательского проекта

Рассчитанная величина затрат научно-исследовательской работы (темы) является основой для формирования бюджета затрат проекта, который при формировании договора с заказчиком защищается научной организацией

в качестве нижнего предела затрат на разработку научно-технической продукции.

Определение бюджета затрат на научно-исследовательский проект приведено в таблице 8.

Таблица 8 - Расчет бюджета затрат НИИ

Наименование статьи	Сумма, руб.
Затраты по основной заработной плате исполнителей темы	303503,76
Отчисления во внебюджетные фонды	91051,13
Затраты на электроэнергию	1373,46
Накладные расходы	39592,84
Бюджет затрат НИИ	435188,19

### **3.3 Определение ресурсной (ресурсосберегающей), финансовой, бюджетной, социальной и экономической эффективности исследования**

Определение эффективности происходит на основе расчета интегрального показателя эффективности научного исследования. Его нахождение связано с определением двух средневзвешенных величин: финансовой эффективности и ресурсоэффективности.

Интегральный финансовый показатель разработки определяется по формуле 11:

$$I_{\text{финр}} = \frac{\Phi_p}{\Phi_{\text{max}}} \quad (11),$$

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

$\Phi_p$  – стоимость исполнения;

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

Расчет:

$$I_{\text{финр}} = \frac{435188,19}{500000} \approx 0,87 \quad (12),$$

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

Полученная величина интегрального финансового показателя разработки отражает соответствующее численное удешевление стоимости разработки в размах.

Интегральный показатель ресурсоэффективности исполнения объекта исследования можно определить по формуле 13:

$$I_p = \sum a * b \quad (13),$$

где  $I_p$  – интегральный показатель ресурсоэффективности;

$a$  – весовой коэффициент;

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

$n$  – число параметров сравнения.

Таблица 9 – Сравнительная оценка характеристик варианта исполнения проекта

Критерии/Объект исследования	Весовой коэффициент параметра	Исп1	Исп2	Исп3	Исп4
1. Способствует росту производительности труда сотрудника компании	0,3	4	4	5	4
2. Удобство в эксплуатации (соответствует требованиям потребителям)	0,25	4	3	5	4
3. Экономия времени	0,2	4	3	5	4
4. Простота исполнения	0,15	5	4	5	4
5. Надежность	0,1	4	4	4	4
<b>Итого:</b>	1				

$$I_{p-исп1} = 4*0,3 + 4*0,25 + 4*0,2 + 5*0,15 + 4*0,1 = 4,15;$$

$$I_{p-исп2} = 4*0,3 + 3*0,25 + 3*0,2 + 4*0,15 + 4*0,1 = 3,55,$$

$$I_{p-исп3} = 5*0,3 + 5*0,25 + 5*0,2 + 5*0,15 + 4*0,1 = 4,9,$$

$$I_{p-исп4} = 4*0,3 + 4*0,25 + 4*0,2 + 4*0,15 + 4*0,1 = 4.$$



Исходя из вычислений, наивысший показатель ресурсоэффективности выполнения имеет вариант исполнения 3.

Интегральный показатель эффективности вариантов исполнения разработки ( $I_{\text{исп}i}$ ) определяется на основании интегрального показателя ресурсоэффективности и интегрального финансового показателя по формуле 14:

$$I_{\text{исп}} = \frac{I_p}{I_{\text{финр}}} \quad (14)$$

Так как для вариантов 1, 2, и 4  $I_{\text{финр}}$  не подсчитывался, примем их равными 1.

$$I_{\text{исп}1} = 4,15$$

$$I_{\text{исп}2} = 3,55$$

$$I_{\text{исп}3} = 4,9 / 0,87 = 5,63$$

$$I_{\text{исп}4} = 4$$

Исходя из расчетов для реализации выбран третий вариант исполнения проекта, так как этот вариант превосходит другие исполнения в ресурсоэффективности.

### **Выводы по разделу**

В ходе работы был определен целевой рынок проекта. В рамках анализа определена трудоемкость данного проекта, она равна 120 дней. На основании трудоемкости была построена диаграмма Ганта.

Так же были рассчитаны величины затрат научно-исследовательских работ. В результате проведенных расчетов, бюджет затрат НИИ составил 435188,19 руб.

После проведенных сравнений и расчетов можно сделать вывод о том, что разработка является бюджетной, так как не превышает запланированный бюджет.

## **4. Социальная ответственность**

### **Введение**

Работа посвящена процессу проектирования и разработки модуля корпоративной информационной системы компании ООО "Умный мир". Разрабатываемый модуль предназначен для заказа обеда в службе доставки "Александра" сотрудниками компании ООО "Умный мир". Для данного модуля будет разработано веб-приложение, а также мобильное приложение, написанное с помощью react-native - фреймворк для разработки кроссплатформенных приложений для IOS и Android.

### **4.1 Правовые и организационные вопросы обеспечения безопасности**

Требования к организации рабочих мест пользователей, обеспечивающие безопасность при работе:

- рабочее место должно быть организовано с учетом эргономических требований согласно ГОСТ 12.2.032-78 «ССБТ. Рабочее место при выполнении работ сидя. Общие эргономические требования» [25] и ГОСТ 12.2.061-81 «ССБТ. Оборудование производственное. Общие требования безопасности к рабочим местам» [26]
- конструкция рабочей мебели (компьютерный стол, офисное кресло, подставка для ног) должна предусматривать возможность регулировки в соответствии с индивидуальными особенностями пользователя для создания комфортных условий для выполнения работы. Вокруг ПК должно быть обеспечено свободное пространство в радиусе как минимум 60-120см

Помещение компании «Умный мир», где выполнялась магистерская работа, имеет следующие характеристики:

- ширина рабочего помещения 8 м, длина – 10 м, высота – 4 м
- площадь – 80 м<sup>2</sup>
- объём помещения – 320 м<sup>3</sup>
- имеется кондиционер, а также естественная вентиляция: двери, окна
- искусственное освещение
- естественное освещение

В данном помещении оборудовано двенадцать рабочих мест, одновременно в работе обычно задействованы 9 человек. Следовательно, в среднем на одного сотрудника приходится не менее 35 м<sup>3</sup> объема помещения и не менее 8 м<sup>2</sup> площади это удовлетворяет требованиям санитарных норм. По санитарным нормам для одного работника должны быть предусмотрены площадь величиной не менее 6 м<sup>2</sup> и объем не менее 24 м<sup>3</sup>, с учетом максимального числа одновременно работающих в смену. Государственными стандартами и правовыми нормами обеспечения безопасности предусмотрена рациональная организация труда в течение смены, которая предусматривает:

- продолжительность рабочей смены, не превышающей 8 часов
- длительность обеденного перерыва не меньше 30 минут
- установление двух регламентируемых перерывов (не меньше 20 минут после 1-2 часов работы и не меньше 40 минут после 2 часов работы)

Обязательно должен быть предусмотрен предварительный медосмотр, который осуществляется при приеме на работу, и периодические медосмотры. Также перед приемом на работу каждый сотрудник должен пройти

инструктаж по технике безопасности, а в дальнейшем с работником должен быть проведен инструктаж по электробезопасности и охране труда.

#### 4.2 Производственная безопасность

Офисные сотрудники подвергаются в основном физическим и психофизиологическим факторам. В таблице 10 представлены все вредные и опасные факторы и их классификация в соответствие с нормативными документами.

Таблица 10. Классификация вредных и опасных факторов

Факторы (ГОСТ 12.0.003- 2015)	Этапы работ			Нормативные документы
	Разраб отано	Изготов лено	Эксплуат ировано	
Отклонение показателей микроклимата	+	+	+	1. СанПиН 2.2.4.548–96. Гигиенические требования к микроклимату производственных помещений. [11]  2. ГОСТ 12.1.005-88 ССБТ. Общие санитарно- гигиенические

				требования к воздуху рабочей зоны. [12]
Недостаточная освещенность рабочей зоны	+	+	+	3. СП 52.13330.2016 Естественное и искусственное освещение. Актуализированная редакция СНиП 23-05-95 [13] 4. СанПиН 2.2.1/2.1.1.1278–03. Гигиенические требования к естественному, искусственному и совмещённому освещению жилых и общественных зданий. [14]
Превышение уровня шума		+		5. ГОСТ 12.1.003-2014 ССБТ. Шум. Общие требования безопасности. [15]

<p>Опасные и вредные производственные факторы, связанные с электромагнитными полями.</p>	<p>+</p>	<p>+</p>	<p>+</p>	<p>6. СанПиН 2.2.4.1340-03 «Гигиенические требования к персональным электронно-вычислительным машинам и организации работы». [16]</p> <p>7. СанПиН 2.2.4.3359-16 "Санитарно-эпидемиологические требования к физическим факторам на рабочих местах" [17]</p> <p>8. ГОСТ 12.1.006-84 ССБТ. Электромагнитные поля радиочастот. Общие требования безопасности. [18]</p>
--	----------	----------	----------	---

5. Психологические факторы	+	+	+	9. ГОСТ 12.0.002-2014 Система стандартов безопасности труда (ССБТ). Термины и определения [19]
6. Повышенное значение напряжения в электрической цепи, замыкание которой может произойти через тело человека	+	+	+	10. ГОСТ 12.4.011-89 ССБТ «Средства защиты работающих. Классификация» [20] 11. ГОСТ 12.0.003-74 ССБТ «Опасные и вредные производственные факторы. классификация» [21]

#### 4.2.1 Отклонение показателей микроклимата

В ГОСТ 12.1.005-88 [12] представлены гигиенические нормативы на параметры микроклимата в рабочем помещении. Микроклимат определяется сочетаниями влажности, температуры воздуха, окружающих поверхностей и скорости движения воздуха действующих на организм человека. В комплексные планы мероприятий по охране труда обязательно включаются мероприятия по оптимизации микроклиматических показателей до



нормативных значений. Согласно СанПиН 2.2.4.548-96 [11] выполняемая работа относится к категории легкая (1б) - интенсивность энергозатрат в пределах 121- 150 ккал/час (140-174 Вт), это работы сидя, стоя или связанные с ходьбой с некоторым физическим напряжением

Таблица 11. Оптимальные величины показателей микроклимата на рабочих местах производственных помещений (СанПиН 2.2.4.548-96 [11])

Период года	Относительная влажность воздуха, %	Скорость движения воздуха, м/с	Температура воздуха, °С	Температура поверхностей, °С
Холодный	60-40	0,1	21 - 23	20 - 24
Теплый	60-40	0,1	23 - 25	22 - 26

Таблица 12. Допустимые величины показателей микроклимата [11]

Период года	Относительная влажность воздуха, %	Скорость движения воздуха, м/с		Температура воздуха		Температура поверхностей, °С
		для диапазона температур воздуха ниже оптимальных величин не более	для диапазона температур воздуха выше оптимальных величин не более	диапазон ниже оптимальных величин	диапазон выше оптимальных величин	
Холодный	15 - 75	0,1	0,2	19,0 - 20,9	23,1 - 24,0	18,0 - 25,0
Теплый	15 - 75	0,1	0,2	20,0 - 21,9	24,1 - 28,0	19,0 - 29,0

В офисе, где велась разработка, температура поверхностей и температура воздуха составляет 20<sup>0</sup>С и 23<sup>0</sup>С соответственно, а влажность воздуха 40%; а в теплый период температура поверхностей и температура воздуха - 24<sup>0</sup>С и 26<sup>0</sup>С соответственно. Сравнивая со значениями из таблицы, отклонений от норм не выявлено.

#### **4.2.2 Недостаточная освещенность рабочей зоны**

Освещение оказывает влияние на общее самочувствие и настроение, определяет эффективность труда. Нерационально организованное освещение может явиться причиной травматизма: недостаточно освещенные опасные зоны, слепящие источники света и блики от них, резкие тени и пульсации освещенности ухудшают видимость и могут вызвать неадекватное восприятие наблюдаемого объекта.

В офисном помещении должно быть, как естественное, так и искусственное освещение. Естественное освещение обеспечивается за счет оконных проемов, коэффициент искусственного освещения (КЕО) которых должен быть не менее 1,2% в местах, где имеется снежный покров и не менее 1,5% на остальной территории. Естественное освещение, свет из окна, должно падать с левой стороны от сотрудника. В офисе установлено одно панорамное окно размером 2,5 на 2,5 метра в наружной стене с регулируемыми жалюзи. Нормируемые показатели естественного, искусственного и совмещенного освещения в соответствии с СанПиН 2.2.1/2.1.1.1278-03 указаны в таблице 13.

Таблица 13. Нормируемые показатели естественного, искусственного и совмещенного освещения [14]

Помещение	Рабочая поверхность и плоскость нормирования КЕО и освещенности и высота плоскости над полом, м	Естественное освещение		Совмещенное освещение	Искусственное освещение					
		КЕО е <sub>н</sub> , %		КЕО е <sub>н</sub> , %	Освещенность, лк			Показатель дискомфорта, М, не более	Коэффициент пульсации освещенности К <sub>п</sub> , %, не более	
		При верхнем или комбинированном освещении	При боковом освещении	При верхнем или комбинированном освещении	При боковом освещении	При комбинированном				При общем освещении
1	2	3	4	5	6	7	8	9	10	
Кабинеты, рабочие комнаты	Г – 0,8	3,0	1,0	1,8	0,6	400	200	300	40	15
Помещение для работы с дисплеями и видеотерминалами, залы ЭВМ	Г – 0,8 Экран монитора: В – 1,2	3,5	1,2	2,1	0,7	500	300	400	15	10

В офисах для организации искусственного освещения рекомендуется применять светильники типа ЛПО36, ЛПО5, ЛПО13, ЛСО4, ЛПО34, ЛПО31 с люминесцентными лампами типа ЛБ. Также допускается применение светильников местного освещения с лампами накаливания. Светильники должны располагаться прямыми или прерывающимися линиями так, чтобы они были параллельны линии зрения сотрудника за компьютером. Защитный угол светильников должен быть не менее 40 градусов. В случае, когда естественного освещения недостаточно, используется общее искусственное

освещение. Основными источниками искусственного освещения используются лампы белого и дневного света ЛБ-20 и ЛД-20.

Произведем расчет освещения производственного помещения. Рассматриваемое помещение имеет светлый цвет потолков и стен, серое покрытие пола. Длина помещения (a) – 10 м., ширина (b) – 8 м., высота (h) – 4 м. В качестве источника света используются светильники, каждый из которых содержит по 4 люминесцентных лампы мощностью 18 Вт.; общая яркость светового потока ( $\Phi$ ) 1150 Лм. Помещение предназначено для работы за персональным компьютером, поэтому нормой освещенности (E) для него согласно СНиП 23-05-95 станет 300 Лк, рабочая плоскость стола находится на расстоянии ( $h_1$ ) 0,8 м. над уровнем пола, коэффициент запаса ( $K_3$ ) равняется 1,4 [3], а приблизительные коэффициенты отражения, согласно таблице 1.9.3 из «Пособие к МГСН 2.06-99 Расчет и проектирование искусственного освещения помещений общественных зданий»: для побеленного потолка – 0,7; для побеленных стен при незанавешенных окнах – 0,5. [27] Сначала находим площадь помещения (S):  $8*10=80$  м<sup>2</sup>. Далее находим индекс помещения по формуле 15:

$$\varphi = \frac{S}{(h-h_1)*(a+b)} = \frac{80}{(4-0,8)*(10+8)} = 1,4 \quad (15)$$

Теперь на основании показателей отражения поверхностей и вычисленного индекса можно из таблицы определить коэффициент использования (U). В данном случае он равняется 57%. [27] Определим необходимое количество светильников по формуле 16:

$$N = \frac{E*S*K_3}{U*n*\Phi_{\text{л}}} = \frac{300*80*1,4}{0,57*4*1150} = 12,8146 \approx 13 \quad (16)$$

В помещении, в котором проводилась работа, используются рядно расположенные потолочные светильники с люминесцентными лампами.

Проведенные расчеты показали, что минимальное число светильников должно быть равно 13. В результате анализа освещенности рабочего места отклонений от норм выявлено не было, так как в помещении находится 18 светильников. Уровень освещенности соответствует нормам в разные периоды светового дня.

#### **4.2.3 Превышение уровня шума**

При постоянном нахождении в помещении где уровень шума более 85 децибел, могут наблюдаться нарушения слуха. Для офиса, в котором велась разработка, основными источниками шума являются расположенные в помещении компьютеры и кондиционер. Уровни шума для различных категорий рабочих мест служебных помещений регламентирует ГОСТ 12.1.003-2014. «ССБТ. Шум. Общие требования безопасности» [15].

Помещения, в которых для работы используют компьютеры не должны соседствовать с помещениями, в которых уровни шума превышают нормируемые значения. В помещениях, которые оборудованы компьютерами, которые являются основным источником шума, уровень шума на рабочем месте должен быть не более 50 дБ.

Рассматриваемый офис по уровню производственных шумов, не выходит за рамки допустимых значений. Уровень шума менее 50 дБ.

#### **4.2.4 Опасные и вредные производственные факторы, связанные с электромагнитными полями.**

Работая за компьютером, сотрудник подвергается воздействию электромагнитного и электростатического полей. Создаваемое персональным компьютером электромагнитное излучение имеет электрическую (Е) и магнитную (Н) составляющие, а также сложный спектральный состав с диапазоном частот от 0 до 1000 МГц. Основным источником

электромагнитных излучений является ЖК монитор, имеющие низкий уровень электромагнитного излучения.

СанПиН 2.2.4.3359-16 определяет нормы допустимых уровней напряженности электрических полей. Они зависят от времени пребывания человека в контролируемой зоне. Время допустимого пребывания в рабочей зоне в часах рассчитывается по формуле  $T=50/E-2$ . Если напряженность электрического поля лежит в диапазоне 20–25 кВ/м, то работа не может продолжаться более 10 минут. При напряженности, не превышающей 5 кВ/м деятельность людей в рабочей зоне может осуществляться в течение 8 часов.

СанПиН 2.2.4.1340-03 «Гигиенические требования к персональным электронно-вычислительным машинам и организации работы», регламентирующий безопасные уровни излучений. В таблицах 14-15 представлены предельно-допустимые уровни напряженности на рабочих местах и допустимые уровни электромагнитных полей.

Таблица 14. Предельно-допустимые уровни напряженности на рабочих местах [17]

Время воздействия за рабочий день, мин	Условия воздействия			
	Общее		локальное	
	ПДУ	ПДУ магнитной	ПДУ	ПДУ магнитной
0-10	24	30	40	50
11-60	16	20	24	30
61-480	8	10	12	15

Таблица 15. Допустимые уровни электромагнитных полей [16]

Наименование параметра	
Напряженность электромагнитного поля на расстоянии 50 см вокруг дисплея до электрической составляющей, В/м, не более:	

в диапазоне частот 5 Гц – 2 кГц	25
в диапазоне частот 2 – 400 кГц	2,5
Плотность магнитного потока на расстоянии 50 см вокруг дисплея, нТл, не более:	
в диапазоне частот 5 Гц - 2 кГц	250
в диапазоне частот 2– 400 кГц	25
Поверхностный электростатический потенциал, В, не более	500

Для снижения уровня излучений проводятся следующие мероприятия:

- применение средств индивидуальной защиты, направленных на экранирование пользователя ПК целиком или отдельных частей его тела;
- употребление профилактических напитков;
- использование других технических средств защиты от электромагнитных излучений;
- сертификация ПК и аттестация рабочих мест;
- применение фильтров и экранов;
- организационно-технические мероприятия.

В рассматриваемом офисе, уровень напряженности электромагнитного поля не превышает предельно-допустимые значения, индивидуальная защита пользователей не требуется.

#### **4.2.5 Психофизиологические факторы**

Во время длительной работы за компьютером сотрудник также подвергается воздействию психофизиологических факторов, таких как эмоциональные перегрузки, умственное перенапряжение, монотонность труда и другие, определены в ГОСТ 12.0.002-2014 [19] и ГОСТ 12.0.003-2015 [22].

Эмоциональные перегрузки способны вызвать изменения функционального состояния центральной нервной системы, что негативно отражается на состоянии организма в целом. Они могут быть вызваны необходимостью выполнения большого объема работы, конфликтными или стрессовыми ситуациями. Умственное перенапряжение может наступать вследствие отсутствия необходимого времени на отдых после продолжительной работы, нарушения режима сна или режима питания. Оно может накапливаться и приводить к возникновению заболеваний.

Отличительными признаками монотонной работы служат однообразие рабочих действий, их многократное повторение и небольшая длительность. Таковой является работа за компьютером. В результате работающий теряет интерес к работе, и у него возникает состояние «производственной скуки». Монотонная работа отрицательно сказывается на эффективности производства: ухудшаются экономические показатели, повышается аварийность, травматизм, растет текучесть кадров. Для снижения эмоциональных перегрузок и умственных перенапряжений предусмотрены перерывы в работе, возможность выбора удобного времени для выполнения работы.

#### **4.2.6 Повышенное значение напряжения в электрической цепи, замыкание которой может произойти через тело человека**

Офис, в котором расположены персональные компьютеры, относится к помещениям без повышенной опасности, так как согласно ГОСТ 12.0.003-74 [21] отсутствуют следующие факторы:

- высокая температура;
- токопроводящая пыль;



- токопроводящие полы;
- сырость;
- возможность одновременного прикосновения человека к имеющим соединение с землёй металлоконструкциям зданий, технологическим аппаратам и механизмам, металлическим корпусам электрооборудования.

Мероприятия, направленные на предотвращение возможности поражения электрическим током, включают в себя следующее:

- при выполнении монтажных работ необходимо использовать только исправно работающий инструмент, аттестованный службой КИПиА;
- заземление корпусов приборов и инструментов, которое поможет защитить от поражения электрическим током, который может возникнуть между корпусом приборов и инструментом при пробое сетевого напряжения на корпус;
- запрет на выполнение работ на задней панели при включенном сетевом напряжении;
- выполнение работ по устранению неисправностей должно производиться компетентными людьми;
- нужно постоянно наблюдать за исправностью электропроводки и в случае обнаружения неисправностей незамедлительно принимать действия по их устранению.

Согласно ГОСТу 12.4.011-89 «Средства защиты работающих. Общие требования и классификация» [20] к средствам защиты от повышенного уровня статического электричества относятся:

- заземляющие устройства;
- нейтрализаторы;
- увлажняющие устройства;
- антиэлектростатические вещества;
- экранирующие устройства.

### **4.3 Экологическая безопасность**

Охрана окружающей среды заключается в устранении отходов жизнедеятельности человека и бытового мусора.

Если персональные компьютеры теряют свою работоспособность, их списывают и отправляют на специализированный склад, на котором уже принимаются меры по утилизации техники и комплектующих. [23]

По статистике вышедшие из строя люминесцентные лампы являются одним из самых распространенных источников ртутного загрязнения. Помимо стекла и алюминия каждая лампа содержит приблизительно 60 мг ртути, поэтому отработавшие люминесцентные лампы являются опасным источником токсичных веществ. [24]

Утилизация таких ламп заключается в их передаче перерабатывающим предприятиям, которые имеют специальное оборудование для переработки вредных ламп в безвредное сырье – сорбент, которое может являться материалом для других производств. Согласно ГОСТ Р 57740-2017 и ГОСТ Р 51768-2001 отработанные люминесцентные лампы относятся к отходам, которые собираются и сортируются отдельно, поэтому их утилизация и хранение должны отвечать определенным требованиям.

#### **4.4 Безопасность в чрезвычайных ситуациях**

Для офиса, в котором ведется разработка, наиболее вероятно возникновение такой ЧС как пожар, который может возникнуть при замыкании электропроводки оборудования, обрыве проводов или же при несоблюдении мер пожарной безопасности. К противопожарным мероприятиям в помещении относятся следующие:

1) помещение должно быть оборудовано средствами тушения пожара, такими как огнетушители, стенд с противопожарным инвентарем, ящик с песком; средствами связи; электрическая проводка осветительных приборов и электрооборудования должна быть в исправном состоянии

2) каждый сотрудник должен знать месторасположение средств тушения пожара и средств связи; знать номера телефонов экстренных служб для оповещения о пожаре; уметь использовать средства пожаротушения

Рассматриваемое помещение оснащено средствами пожаротушения в соответствии с нормами:

- 1) огнетушитель пенный ОП-10 – 1 шт
- 2) огнетушитель углекислотный ОУ-5 – 1 шт

В помещении и на этаже присутствуют следующие средства оповещения:

- световая индикация направления движения к выходу в коридорах этажа;
- звуковая индикация, которая представляет собой систему оповещения о пожаре через громкоговоритель;
- пассивные датчики задымленности.

Чтобы минимизировать вероятность возникновения пожара нужно своевременно проводить профилактические работы, направленные на устранение возможных источников возникновения пожара, такие как:

- систематическое наблюдение за состоянием электропроводки;
- выключение питания оборудования при завершении работы и покидании рабочего места;
- периодическое проведение инструктажа по пожаробезопасности для персонала.

При возникновении пожара должна сработать система пожаротушения, издав предупредительные сигналы, и передав на пункт пожарной станции сигнал о ЧС, в случае если система не сработала, по каким-либо причинам, необходимо нажать тревожную кнопку или самостоятельно произвести вызов пожарной службы по телефону 101, сообщить место возникновения ЧС и ожидать приезда специалистов.

### **Выводы по разделу**

В данном разделе был произведен анализ рабочего помещения, анализ вредных и опасных факторов и методы минимизации их воздействия на человеческое здоровье. Были рассмотрены нормативы микроклимата, освещения, шума, электробезопасности помещения. Исходя из полученных данных можно утверждать, что оно соответствует требованиям законодательства РФ. Также были рассмотрены аспекты экологической, производственной безопасности, безопасности в чрезвычайных ситуациях (на примере пожароопасности).

## ЗАКЛЮЧЕНИЕ

Таким образом, в ходе выполнения магистерской диссертации был спроектирован и разработан web-сервис кейтеринговых услуг для сотрудников компании ООО «Умный мир». В ходе работы был произведен обзор используемых технологий и обоснование их выбора.

Мобильное приложение разрабатывалось на языке javascript с использованием фреймворка React native. Веб-приложение также разрабатывалось на языке javascript с использованием фреймворка Vue JS. Серверная часть разрабатывалась на языке PHP с использованием фреймворка Laravel. Также использовался сервис очередей Redis. Для хранения данных использовалась СУБД PostgreSQL. Для отслеживания изменений и хранения проекта в удалённом хранилище использовался Gitlab.

Web-сервис отвечает всем необходимым по заданию требованиям, но в данный момент находится в стадии тестирования, поэтому пока ещё не удалось опубликовать мобильное приложение в PlayMarket и AppStore, что будет сделано в дальнейшем.

## **Conclusion**

Thereby, in the course of this master's thesis there was developed a web service for the employees of "Smart World" LLC. In the course of the work it was also made a review of the used technologies as also a provision of a rationale for their choice.

The mobile application was developed in javascript with using of the React native framework. The web application was also developed in javascript with using of the Vue JS framework. The server part was developed in the PHP language with using of the Laravel framework. There was also used Redis queue service. For data storage it was used the PostgreSQL DBMS. It was also used Gitlab, as well for changes tracking, as also for the project storage on a remote data warehouse.

The web service meets all the requirements that were defined in the course of work to accomplish all the set tasks, but is currently in the testing phase, so it has not yet been possible to publish the mobile application on PlayMarket and AppStore, which is going be done in process of time.

## СПИСОК ИСТОЧНИКОВ

1. Основы Git [Электронный ресурс] – Режим доступа <https://git-scm.com/book/ru/v2/Введение-Основы-Git> (дата обращения 05.04.2020).
2. Документация React Native [Электронный ресурс] – Режим доступа <https://reactnative.dev/docs/getting-started> (дата обращения 01.04.2020).
3. Документация React JS [Электронный ресурс] – Режим доступа <https://ru.reactjs.org/docs/getting-started.html> (дата обращения 03.04.2020).
4. Руководство Vue JS [Электронный ресурс] – Режим доступа <https://ru.vuejs.org/v2/guide/> (дата обращения 12.04.2020).
5. Документация Vue Native [Электронный ресурс] – Режим доступа <https://vue-native.io/docs/> (дата обращения 13.04.2020).
6. Документация Laravel [Электронный ресурс] – Режим доступа <https://laravel.com/docs/7.x> (дата обращения 15.04.2020).
7. Что такое кейтеринг – услуги, форматы обслуживания [Электронный ресурс] – Режим доступа <https://vkusnoff-catering.ru/articles/chto-takoe-keytering/> (дата обращения 01.04.2020).
8. Язык UML. Руководство пользователя / Гради Буч, Джеймс Рамбо, Ивар Якобсон – ДМК Пресс, 2007.
9. Документация к PostgreSQL [Электронный ресурс] – Режим доступа <https://www.postgresql.org/docs/>.
10. Базы данных. Учебное пособие / И.Л. Чудинов, В.В. Осипова – Издательство Томского политехнического университета, 2011.
11. СанПиН 2.2.4.548–96. Гигиенические требования к микроклимату производственных помещений.
12. ГОСТ 12.1.005-88 ССБТ. Общие санитарно-гигиенические требования к воздуху рабочей зоны.

13. СП 52.13330.2016 Естественное и искусственное освещение. Актуализированная редакция СНиП 23-05-95
14. СанПиН 2.2.1/2.1.1.1278–03. Гигиенические требования к естественному, искусственному и совмещённому освещению жилых и общественных зданий.
15. ГОСТ 12.1.003-2014 ССБТ. Шум. Общие требования безопасности.
16. СанПиН 2.2.4.1340-03 «Гигиенические требования к персональным электронно-вычислительным машинам и организации работы»
17. СанПиН 2.2.4.3359-16 "Санитарно-эпидемиологические требования к физическим факторам на рабочих местах"
18. ГОСТ 12.1.006-84 ССБТ. Электромагнитные поля радиочастот. Общие требования безопасности.
19. ГОСТ 12.0.002-2014 Система стандартов безопасности труда (ССБТ). Термины и определения
20. ГОСТ 12.4.011-89 ССБТ «Средства защиты работающих. Классификация»
21. ГОСТ 12.0.003-74 ССБТ «Опасные и вредные производственные факторы. Классификация
22. ГОСТ 12.0.003-2015 ССБТ. Опасные и вредные производственные факторы. Классификация
23. ГОСТ Р 57740-2017 Ресурсосбережение. Обращение с отходами. Требования к приему, сортировке и упаковыванию опасных твердых коммунальных отходов
24. ГОСТ Р 51768-2001. Ресурсосбережение. Обращение с отходами. Методика определения ртути в ртутьсодержащих отходах.



25. ГОСТ 12.2.032-78 ССБТ. Рабочее место при выполнении работ сидя. Общие эргономические требования.

26. ГОСТ 12.2.061-81 ССБТ. Оборудование производственное. Общие требования безопасности к рабочим местам.

27. «Пособие к МГСН 2.06-99 Расчет и проектирование искусственного освещения помещений общественных зданий» [Электронный ресурс] – Режим доступа <https://meganorm.ru/Data2/1/4294846/4294846582.htm> (дата обращения 11.05.2020)

## Приложение А

### Design and implementation of software modules

Студент

Группа	ФИО	Подпись	Дата
8ВМ81	Гузиков Даниил Родионович		

Руководитель ВКР

Должность	ФИО	Ученая степень, звание	Подпись	Дата
Доцент ОИТ	Цапко С.Г.	к.т.н.		

Консультант – лингвист отделения иностранных языков ШБИП

Должность	ФИО	Ученая степень, звание	Подпись	Дата
Доцент ШБИП	Сидоренко Т.В.	к.т.н.		

## **1. Research of the subject area**

### **1.1 Description of the subject area**

The term "Catering" comes from the English word "cater", which means "food supply." The derivative word "catering" means foodservice industry. This term is understood as the provision of food services for employees by contracting companies, as well as restaurant services outside of stationary rooms, as well as food services at various outdoor events. In practice, catering does not only mean the delivery and preparation of food. This concept also includes service, room decoration, table setting and other similar services.

The company's catering expenses are beneficial not only for employees, but also for the employer, as company employees are more motivated to work fruitfully. Therefore, their productivity is increasing as the company's value in the labor market.

### **1.2 A brief overview of web service architecture for catering services and technologies**

It was decided to use a three-tier architecture for the web service, which includes the following levels:

- Client: this level should not have direct connections with the database (for security and scalability requirements), be loaded with basic business logic (for scalability requirements) or store the state of the application (for reliability requirements). Only the simplest business logic is usually taken to this level: the authorization interface, encryption algorithms, checking the entered values for validity and format compliance, simple operations with data (sorting, grouping, calculation of values) that are already loaded on the terminal.
- Application Server is located on the second level, the main part of the business logic is concentrated on it. Outside of it remain only fragments exported to the client (terminals), as well as logic elements included in the database (stored

procedures and triggers). The implementation of this component is provided by middleware. Application servers are designed in such a way that adding additional instances to them ensures horizontal scaling of the software package performance and does not require any changes in the application program code.

- Database server: provides data storage and locates on a separate level. As a rule, it is implemented by the tools of database management systems; the connection to this component is provided only from the application server level.

The used three-tier web service architecture is schematically shown in Figure 1.

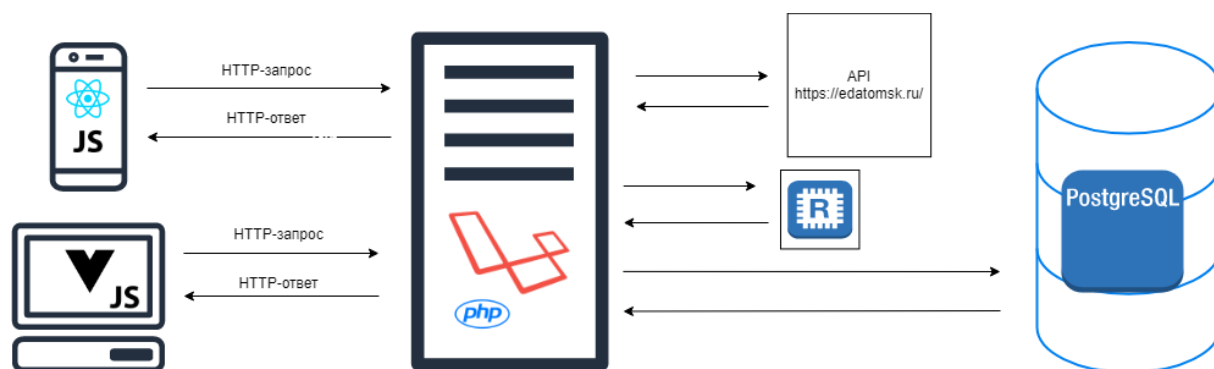


Figure 1 – The used architecture

It was used PostgreSQL to organize data storage.

PostgreSQL is a universal and object-relational database management system with an open source code. PostgreSQL was developed on the basis of POSTGRES 4.2 at the University of California, Berkeley.

Initially, the idea of creating PostgreSQL was to use it on UNIX-like platforms. But subsequently PostgreSQL has also been developed for other platforms as well, such platforms as the Mac OS X, Solaris and Windows. This DBMS is free, which allows you to freely use, modify and distribute PostgreSQL in any form.

PostgreSQL requires minimal effort due to its stability. Consequently, the development of an application based on PostgreSQL can reduce total ownership cost compared to other database management systems.

To develop the web server was used the PHP language, which is a server-side scripting language. It is commonly used to develop static or dynamic websites and web applications. This language has many frameworks, such as Symphony, Laravel, Yii2, Zend. For this work the choice has been made in favor Laravel, a description of the capabilities of which will be given in the paragraph 2.2.

Also, the web server interacts with the Alexandra delivery service API to get the data on the current menu.

Redis was used as a server for storing queues. The queue allows you to postpone the task for some period, usually used to send an e-mail or generate pdf files to users.

For the development of the web client and mobile application was used JavaScript, which is an object-oriented programming language. This language also has many frameworks for developing of web applications and mobile applications. For example, such frameworks as React JS, Vue JS, Angular, etc., which are described in paragraph 2.3, are quite popular in the area of development of web applications. So that frontend developers could also develop interfaces for mobile applications, there were created such frameworks as React native and Vue native, the capabilities of which will be described in the paragraph 2.4.

## **2. Design and implementation of software modules**

### **2.1 Requirements and task setting**

A developed web-service for catering services must meet the following requirements:

- The web-service should receive data on the current menu in the delivery service "Alexandra" for a week in advance.

- Each employee in the company can fill the shopping cart with a daily limit of 200 rubles. If the limit is exceeded, the user must be informed about it.
- The mobile application must have a cross-platform UI (Android and IOS versions).
- The web application must be supported by all modern browsers (Firefox, Chrome, Safari, Opera).
- It has to be possible to blacklist a dish if it does not satisfy the user.
- After an order has been made, the system must generate a csv file according to the template defined by the Alexandra delivery service and send it to its E-mail address. This file should contain the information about the orders of all employees for the next day.

On the basis of these requirements we have set following tasks. After they are performed the web-services is considered as a minimal viable product (all tasks given below are both relevant for the web version and mobile app)

- To implement an employee authorization service;
- To implement the reception of data on the current menu from the server for the next 7 days;
- To develop a shopping cart module that will track the daily limit of 200 rubles per employee;
- To develop an algorithm for adding a dish to the Black List;
- To organize the storage of data on employees orders in the database, as well as the creating of a csv file containing a table with the orders of all employees and sending this file to the mail address of the Alexandra delivery service;
- To develop and implement an equable design for the mobile application and for the web client;
- To test the web client and the mobile application with automated tests and a testing team in the company;
- To publish a mobile application on PlayMarket and AppStore.

## 2.2 Selection of software and programming technology for server side

For the development of the server part, was chosen the PHP language and the Laravel PHP framework.

Laravel is a PHP framework with an open source code which is used for the web development. It is based on the Symfony platform which is a collection of reusable components and libraries, which makes it easier to develop high-performance applications.

The framework provides the developer with built-in components for frequently used web application tasks, such as Network authentication, caching, sessions, which might be its biggest advantage.

The Benefits of Laravel:

- Storage - a separate place in the storage structure.
- Eloquent - ORM for working with the database and implementation of the ActiveRecord template. In addition to standard CRUD operations, it also includes: access methods and mutators, secure deletion, relationships and an observer model.
- Artisan - allows you to manage the command line.
- Simple routing, checking incoming parameters.
- Integrated SCSS and Script Collector.
- Built-in template engine Blade.
- The framework secures an efficient API for a quick sending of e-mails (useful when sending the order by Email).
- Task Queues, Task Scheduler.
- Has excellent database migration systems.
- Has an integrated unit testing system.
- Support for web-sockets for developing of interactive applications.
- ACL Plugins.

- High popularity across the West.
- Frequent releases of new versions.

Among its weaknesses can be marked a lack of built-in interface templates.

The main feature of this framework is a proper database architecture. During development it is recommended to cover each of the available routes with tests, fill the database with test data and follow the basis of the Laravel's philosophy, which is brevity.

Laravel gives the developer a great freedom in choosing solutions and ways of their implementation. This is both good and bad. On the one hand, it makes the programming process easier, on the other hand, it increases the labor costs from the part of development.

### **2.3 Selection of software tools and programming technology for the web client**

Since the developer has some experience with React JS and Vue JS, the choice of a framework for developing the web client was being made between these two options that are suitable for solving the set tasks.

React and Vue are very similar in their approach to the work with the DOM (document object model). They both use the Virtual DOM approach to the elements rendering and the re-rendering of elements in the browser. The frameworks that use this approach store a virtual DOM copy of the browser. Then they can use this copy to determine how to display the new changes.

Although React and Vue use the same approach to the DOM, the way they render HTML and CSS is different.

Let us start with how React handles the rendering of elements on a web page. React does this with components. A basic React component is given in the attachment B.



The React's approach to HTML and CSS is based on the usage of JSX allows developers to define their HTML templates (and often the CSS rendering too) in Javascript files. The CSS processing can be performed by various libraries (e.g. StyledComponents).

In our example, we used the ReactDOM class to find the entry point (in our case, a div with the identifier "root") and to render the App component.

It is also possible to add more components to our React application, for that they must be embedded in the component template. This example is also given in the attachment B.

Vue also uses a component approach to render HTML and CSS code in the browser. This means that this framework is slightly different from React. Vue's approach is to use HTML templates to determine how components are rendered. This example is also given in the attachment B.

Vue's approach to the HTML and CSS rendering relies more on actual HTML templates to determine how the components are structured and ordered. This does not completely exclude templates from the Vue component file. The HTML sections still must be defined in our component. In this example, all the code is processed by Javascript.

Vue also allows you to use the concept of Single File Components. This approach allows you to include CSS styles, templates and the component logic in a single file. Single-file components offer efficient tools for incorporating of CSS without a need to install a special library above Vue. However, if you have a CSS library that you would like to use, Vue will most likely support it. Libraries such as Vue Loader help developers to use CSS modules in their Vue components.

Although Vue has HTML a default templating, it also supports JSX templates through the Babel extension. The way Vue works with JSX is different

from React, but gives developers the possibility to use Vue without giving up on JSX.

## **2.4 Selection of software and programming technology for a mobile application**

Since the developer has some experience with the React JS and the Vue JS, the choice of a framework for the development of the mobile application was being made between JS-frameworks React Native and Vue Native, which are suitable for the set tasks.

What is React Native? It is a framework for developing your own mobile apps using React. React Native allows you to create world-class applications on your own platforms, using the coordinated capabilities of developers based on JavaScript and React. The focus of React Native is the productivity of developers on all platforms that interest you – study once, write anywhere. Facebook uses React Native in several production applications and will continue to invest in React Native.

What is Vue Native? It is the development of cross-platform mobile applications using Vue.js. Vue Native is a mobile platform for creating a truly native mobile application using Vue.js. Vue Native is a shell for React Native API interfaces that allows you to use Vue.js and create a great mobile user interface.

For the development of the mobile application, the choice has been made in favor of the React Native, since, according to the generally accepted opinion, Vue Native is still "a bit raw" for production versions, in contrast to React Native. Our company also has several other projects where the framework react-native has proved its worth.

From official documentation of React native, it becomes known that there are two ways to launch a developed application: React native init (CLI) or Expo. The comparative analysis of these methods is given in the table 1.

Table 1 – Comparative analysis of React native init and Expo

Name	React native init(CLI)	Expo
Adding Java/Objective-C modules	Yes	No
Weight of the standard “Hello World” application	5 mb	25 mb
Requires Android Studio and Xcode to run projects	Yes	No
Fonts must be manually sorted in Xcode	Yes	No
Share the app through .apk	Yes	No(QR-code)
The ability to open a project while working on it	No	Yes

Due to the lower weight of a standard application, as well to the possibility to share the application through .apk, the choice has been made in favor of react native init(CLI).

## Приложение Б

Примеры компонентов для различных фреймворков:

- Базовый компонент React:

```
class Welcome extends React.Component {  
  render() {  
    return <h1>Hello</h1>;  
  }  
}
```

- Пример компонентного подхода в React

```
render() {  
  return (  
    <div>  
      <MyComponent />  
      <MyOtherComponent />  
    </div>  
  )  
}
```

- Пример компонента Vue

```
Vue.component('button-counter', {  
  data: function () {  
    return {  
      count: 0  
    }  
  },  
});
```

```
template: '<button v-on:click="count++">Счётчик кликов — {{ count  
}}</button>'  
))
```