

МИНИСТЕРСТВО ОБРАЗОВАНИЯ И НАУКИ РЕСПУБЛИКИ КАЗАХСТАН  
Некоммерческое акционерное общество  
«АЛМАТИНСКИЙ УНИВЕРСИТЕТ ЭНЕРГЕТИКИ И СВЯЗИ  
им. ГУМАРБЕКА ДАУКЕЕВА»  
Кафедра IT – инжиниринг

«ДОПУЩЕН К ЗАЩИТЕ»  
Зав. кафедрой PhD, доцент Досжанова А.А  
\_\_\_\_\_ «\_\_\_\_» \_\_\_\_\_ 2020 г.

**ДИПЛОМНЫЙ ПРОЕКТ**

На тему: Разработка подсистемы непрерывного управления рисками для системы поддержки бизнеса компании

Специальность 5B070400 – Вычислительная техника и программное обеспечение

Выполнила: Нурлан Е.Н           Группа ВТ-16-2

Научный руководитель: PhD, доцент Досжанова А.А

Консультанты:

по экономической части: к.э.н., профессор Габелашвили К.Р

(учёная степень, звание, Ф.И.О.)

\_\_\_\_\_ «\_\_\_\_» \_\_\_\_\_ 2020 г.

по безопасности жизнедеятельности: доцент Приходько Н.Г

(учёная степень, звание, Ф.И.О.)

\_\_\_\_\_ «\_\_\_\_» \_\_\_\_\_ 2020 г.

по программному обеспечению: ст.преп. Майкотов М.Н

(учёная степень, звание, Ф.И.О.)

\_\_\_\_\_ «\_\_\_\_» \_\_\_\_\_ 2020 г.

Нормоконтролер: ст.преп. Абсатарова Б.Р

(учёная степень, звание, Ф.И.О.)

\_\_\_\_\_ «\_\_\_\_» \_\_\_\_\_ 2020 г.

Рецензент: \_\_\_\_\_

(учёная степень, звание, Ф.И.О.)

\_\_\_\_\_ «\_\_\_\_» \_\_\_\_\_ 2020 г.

Алматы 2020

МИНИСТЕРСТВО ОБРАЗОВАНИЯ И НАУКИ РЕСПУБЛИКИ КАЗАХСТАН  
Некоммерческое акционерное общество  
«АЛМАТИНСКИЙ УНИВЕРСИТЕТ ЭНЕРГЕТИКИ И СВЯЗИ  
им. ГУМАРБЕКА ДАУКЕЕВА»

Институт систем управления и информационных технологий

Кафедра IT-инжиниринг

Специальность 5В070400 – «Вычислительная техника и программное обеспечение»

**ЗАДАНИЕ**

на выполнение дипломного проекта

Студенту Нурлан Елдосу Нурланулы

Тема проекта: Разработка подсистемы непрерывного управления рисками для системы поддержки бизнеса компании

Утверждена приказом по университету № \_\_\_ от «\_\_\_» \_\_\_\_\_ 2020 г.

Срок сдачи законченного проекта «\_\_\_» \_\_\_\_\_ 2020 г.

Исходные данные к проекту (требуемые параметры результатов исследования (проектирования) и исходные данные объекта): руководство по программированию на PHP, интегрированная среда Plesk, данные преддипломной практики.

Перечень вопросов, подлежащих разработке в дипломном проекте, или краткое содержание дипломного проекта:

- а) анализ и исследование предметной области;
- б) проектирование приложения;
- в) реализация приложения;
- г) экономическое обоснование эффективности;
- д) вопросы безопасности жизнедеятельности и охраны труда.

Перечень графического материала (с точным указанием обязательных чертежей): имеется 18 таблиц, 62 иллюстрации и 1 диаграмма.

Основная рекомендуемая литература:

1 Никсон, Робин Создаем динамические веб-сайты с помощью PHP, MySQL, JavaScript и CSS / Робин Никсон. - М.: "Издательство "Питер", 2013. - 560 с.

2 Прохоренок, Николай HTML, JavaScript, PHP и MySQL. Джентльменский набор Web-мастера (+ CD-ROM) / Николай Прохоренок. - М.: БХВ-Петербург, 2012. - 912 с.

3 Дронов В. PHP, MySQL и Dreamweaver. Разработка интерактивных Web-сайтов; БХВ-Петербург - М., 2016. - 480 с.

Консультация по проекту с указанием относящихся к ним разделов проекта

Раздел	Консультант	Сроки	Подпись
Экономическая часть	Габелашвили К.Р	24.04.2020	
Безопасности жизнедеятельности	Приходько Н.Г	24.04.2020	
Программная часть	Майкотов М.Н	15.05.2020	
Нормконтролер	Абсатарова Б.Р	18.05.2020	

**ГРАФИК**  
подготовки дипломной работы (проекта)

Наименования разделов, перечень разрабатываемых вопросов	Сроки представления научному руководителю	Примечания
Анализ и исследование предметной области	17.02.2020 - 26.02.2020	
Проектирование приложения	27.02.2020 - 23.03.2020	
Программная реализация	24.03.2020 - 20.04.2020	

Дата выдачи задания « \_\_\_\_ » \_\_\_\_\_ 2020 г.

Заведующий кафедрой \_\_\_\_\_ А.А.Досжанова  
(подпись)

Научный руководитель проекта \_\_\_\_\_ А.А.Досжанова  
(подпись)

Задание принял к исполнению студент \_\_\_\_\_ Н.Е.Нурлан  
(подпись)

## Аңдатпа

Дипломдық жобаның тақырыбы: «Компанияның бизнесін қолдау жүйесі үшін тәуекелдерді үздіксіз басқарудың қосалқы жүйесін құру».

Бұл дипломдық жобада HoReCa сегменті үшін азық-түлікке тапсырыс берудің және сатудың автоматтандырылған жүйесін әзірлеу жүзеге асырылды, оның мақсаты компанияның тәуекелдерін азайту және Қазақстан қалаларында азық-түлікке тапсырыс беру сервисін құру болып табылады, онда мейрамханалардың иелері жоғары деңгейдегі сервисті және пайдалану ыңғайлылығын алады, ал жеткізушілер өз өнімдерін сатудың қосымша арнасын алады.

Дипломдық жоба web - платформа форматында орындалды. HTML, CSS, PHP, JavaScript сияқты технологиялар әзірленген дипломдық жобаның мақсатына жету үшін пайдаланылады.

Сонымен қатар, бағдарламаны әзірлеудің шығындары мен бағасының қаржылық есебі, әзірленетін жобаның қаржылық қажеттілігін бағалау жүргізіледі және өнеркәсіптік жарықтандырудың сапасын арттыру бойынша іс-шаралар ұсынылады.

## **Аннотация**

Тема дипломного проекта: «Разработка подсистемы непрерывного управления рисками для системы поддержки бизнеса компании».

В данном дипломном проекте была реализована разработка автоматизированной системы заказа и продаж продуктов питания для сегмента HoReCa, цель которого состоит в том, чтобы минимизировать риски компании и создать сервис заказа продуктов питания в городах Казахстана, где владельцы ресторанов получают сервис высокого уровня и удобство использования, а поставщики получают дополнительный канал продаж своей продукции.

Дипломный проект был выполнен в формате web - платформы. Такие технологии, как HTML, CSS, PHP, JavaScript были использованы для достижения цели разрабатываемого дипломного проекта.

Кроме того, был проведен финансовый расчет издержек и цены разработки программы, оценка финансовой необходимости разрабатываемого проекта и были предложены мероприятия по повышению качества промышленного освещения.

## **Abstract**

The theme of the diploma project: «Development of a continuous risk control subsystem for a company's business support system».

This diploma project included the development of an automated food ordering and sales system for the HoReCa segment, which aims to minimize the company's risks and create a food ordering service in the cities of Kazakhstan, where restaurant owners will receive a high-level service and usability, and suppliers will receive an additional sales channel for their products.

The diploma project was made in the format of a web platform. Technologies such as HTML, CSS, PHP, and JavaScript were used to achieve the goal of the diploma project being developed.

In addition, a financial calculation of the cost and price of developing the program, an assessment of the financial need for the project being developed, and measures to improve the quality of industrial lighting were proposed.

## Содержание

Введение .....	8
1 Обзор и анализ предметной области .....	10
1.1 Эффективность внедрения информационных систем .....	10
1.2 Анализ рынка HoReCa .....	13
1.3 Анализ существующих IT решений рынка общественного питания .....	14
1.4 .....	Постановка задачи.....15
2 Проектирование системы и обоснование выбора инструментов .....	17
2.1 Проектирование закупочной системы общепита .....	17
2.2 Выбор инструментов для разработки платформы .....	21
3 Разработка программного продукта.....	25
3.1 Системные требования.....	25
3.2 Описание разработанных функций .....	25
3.3 Описание клиентской части.....	38
3.4 Создание отдельной страницы для маркетинговой стратегии .....	49
4 Экономический расчет .....	51
4.1 Цели и задачи, решаемые в экономической части .....	51
4.2 Расчет трудоемкости разработки ПП .....	51
4.3 Расчет затрат на разработку ПП .....	51
4.4 Расчет затрат на оплату труда .....	53
4.5 Определение возможной(договорной) цены ПП .....	57
4.6 Сравнительный анализ эксплуатационных затрат до и после внедрения .....	58
5 Безопасность жизнедеятельности.....	63
5.1 Анализ потенциально опасных и вредных факторов в офисе, воздействующих на персонал .....	63
5.2 Расчет естественного освещения.....	66
5.3 Расчет искусственного освещения .....	68
5.4 Расчет систем кондиционирования рабочего помещения IT отдела.....	71
Заключение .....	76
Список литературы .....	77
Приложение А Техническое задание .....	79
Приложение Б Листинг программы .....	81
Приложение В Акт внедрения.....	99

## Введение

Казахстанский рынок общественного питания оценивается в 70 миллиардов долларов США, тем самым расположив его после нефтяного рынка. Сама же структура рынка схожа по всему миру, где крупные торговые дома и дилеры покупают продукты у экспортирующих стран и перепродают производителям, а дальше рестораны готовят из него свои блюда.

Актуальность темы заключается в том, что в последнее время рестораторы стали больше внимания уделять автоматизации своих предприятий. Этому требуют современные тенденции ресторанного рынка, растущая конкуренция, желания ресторатора сделать свой бизнес более прозрачным и легким в управлении.

В последнее время рестораторы и отельеры начали уделять огромное внимание автоматизации своих ресторанов. В нынешних условиях наибольшей популярностью пользовались IT-системы, которые позволяют автоматизировать самые основные функции – обычно это продажи и склад. Кризис же заставил задуматься о том, как «выжать» из компании максимальную эффективность, сохранить постоянных гостей, получить необходимые оборотные средства на поддержание и развитие бизнеса. Современная IT-платформа, позволяющая едино управлять всеми ресурсами компании – это то, что поможет не просто держаться на плаву, но и обеспечить потенциал для развития бизнеса и снизить его риски.

В наши дни можно с уверенностью говорить о том, что автоматизация – это необходимость 21 века и среди владельцев ресторанно-гостиничного бизнеса есть полное понимание, на чем можно сэкономить, а без чего невозможно организовать стабильную работу предприятия.

Профессиональная система автоматизации - это надежный инструмент, необходимый для постоянного мониторинга и отслеживания деятельности учреждения. Это позволяет быстро и эффективно обслуживать клиентов, оптимизировать работу с поставщиками, наладить прозрачное управление, бухгалтерский и финансовый учет, устранить злоупотребления, своевременно принимать ответственные решения и многое другое.

Во времена кризиса количество гостей в ресторане может быть значительно сокращено. Чтобы привлечь посетителей, придется снизить цены, в то время как бизнес должен будет существовать с наценкой, которая будет ниже, чем раньше. Выживание в этих условиях поможет контролировать цены поставщиков на входе, точное управление затратами, оптимизацию уровней запасов на складах, составление бюджета (чтобы не попасть в дефицит наличности). В современном мире самые дальновидные рестораторы будут разумно рассуждать и внедрять современные системы управления для своей компании.

Целью дипломного проекта является – создание автоматизированной системы заказов и продаж продуктов питания для сегмента HoReCa, а также



анализ и минимизация рисков для поставщиков, ресторанов, отелей и сетей быстрого питания.

В ходе работы были использованы методы исследования, такие как: интервьюирование ресторанов и поставщиков нашего города, изучение и обобщение отечественной и зарубежной практики, а также для разработки информационной системы был проведен анализ литературы.

Основным преимуществом системы автоматизации является возможность быстро получать практически всю информацию, необходимую для принятия решений, а также отслеживать динамику прибыли и рассчитывать производственные затраты, контролировать и пресекать случаи краж и злоупотреблений со стороны персонала, поддерживать историю работы с поставщиками, организация и проведение дисконтных программ в их учреждениях для привлечения клиентов и многое другое.

Апробация разработанной системы проводилась в ТОО «RESTOFOOD(РЕСТОФУД)» города Алматы и подтверждается актом внедрения.

## **1 Обзор и анализ предметной области**

### **1.1 Эффективность внедрения информационных систем**

По источникам из открытых данных, 30% потерь от выручки - это очень большая цифра. Если средняя прибыль ресторана в день составляет 400 тысяч тенге, получается, ежедневно хозяева недополучают прибыль в размере около 120 тысяч тенге.

Основными причинами внедрения системы автоматизации являются: злоупотребление персоналом (хищения), необходимость повышения качества и скорости обслуживания клиентов, как за счет повышения конкурентоспособности, так и проницаемости (рентабельности), отчетность о результатах деятельности, возможность проведения ценообразования, маркетинговой политики и единой номенклатуры из головного офиса [1].

Основной проблемой, с которой сталкиваются не только владельцы магазинов, но и владельцы кафе или ресторанов, является кража сотрудников. По оценкам экспертов, потери составляют в среднем 30% выручки. Спасение от этого - внедрение эффективной системы автоматизации.

Во-первых, это связано с отсутствием оперативной информации о сумме выручки менеджеров. Поэтому сотрудники и обмениваются «кейтеринговыми» товарами самостоятельно, закупаются где-то за дешевые, недостаточные или обесцененные покупателем товары.

Во-вторых, кража товаров сотрудниками ресторана может происходить из-за отсутствия управленческой информации в инвентаре. Наконец, прямая потеря денег или продуктов часто означает, что ответственные средства не контролируются.

Конечно, даже самая совершенная система автоматизации не решит проблему кражи на 100%. Однако при правильном применении она может помочь минимизировать потери, доведя их до 3-5%. Сама по себе автоматизация – это эффективный инструмент управления учреждением и особенно контроля персонала.

В дополнение к системе автоматизации, в компании общественного питания есть еще один способ предотвращения крупных краж [2]. Официанты принимают заказы от покупателей, выставляют им счета, но не получают деньги, а только кассир на выходе из магазина. Эта система очень проста, но, конечно, ее трудно использовать в реальных условиях. Между тем официанты в большинстве ресторанов получают деньги, изобретая сотни способов их украсть.

Вторая причина, помимо кражи, которая заставляет владельцев баров и ресторанов задуматься об автоматизации, заключается в том, чтобы получать отчеты о работе учреждения: всегда быть в курсе событий и знать, как идут дела. Используя хорошее программное обеспечение, менеджмент может решить, какие изменения внести в ресторан. Например, если алкоголь хорошо

продается в заведении, но мясные блюда вообще не идут, могут потребоваться некоторые модификации и закупки большого количества алкоголя.

Третья причина заключается в том, что ресторан или бар извлекает выгоду из автоматизации, соответственно улучшая и обновляя сервис. В автоматизированном заведении общественного питания покупатель вряд ли перепутает заказ, который он получает от самого ресторана. Обслуживание клиентов через систему ускоряется. В результате посетители ресторана или кафе получают только самые свежие, самые качественные продукты. Кроме того, управление предприятиями общественного питания может снизить производственные затраты за счет оптимизации запасов. Благодаря программе вы также можете отслеживать продукты, чтобы отслеживать, заканчивается ли конкретный продукт.

В настоящее время рынок программных продуктов для автоматизации отрасли HoReCa достаточно широк. Основное место на нем занимают разработчики, которые вовремя сумели предложить рестораторам продуманные, адаптированные для казахстанского бизнеса и более дешевые решения по сравнению с западными [3].

Основная конкуренция между разработчиками происходит между решениями на основе «1С» и «оригинальными» продуктами. Как правило, решения 1С больше ориентированы на казахстанских рестораторов, поскольку они предлагают не только обычный оперативный складской учет, но и финансовый учет (бухгалтерский, налоговый), при котором использование продуктов 1С стало стандартом.

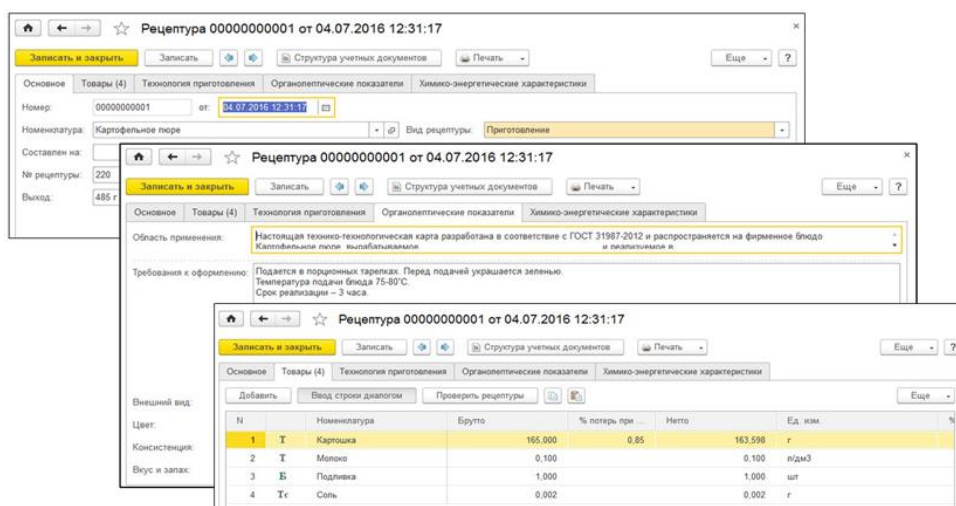


Рисунок 1.1.1 – Пример решения «1С»

Помимо отечественных разработчиков на рынке присутствуют программные продукты зарубежных компаний. Доли этих западных брендов в Казахстане невелики, несмотря на их популярность за рубежом. Как правило, эти системы не являются специализированными для сектора общественного питания, но требуют дополнительной настройки для конкретной компании. В

конце концов, можно обнаружить, что программный продукт ориентирован не на бизнес, а наоборот, бизнес должен быть адаптирован к конкретному программному продукту [1].

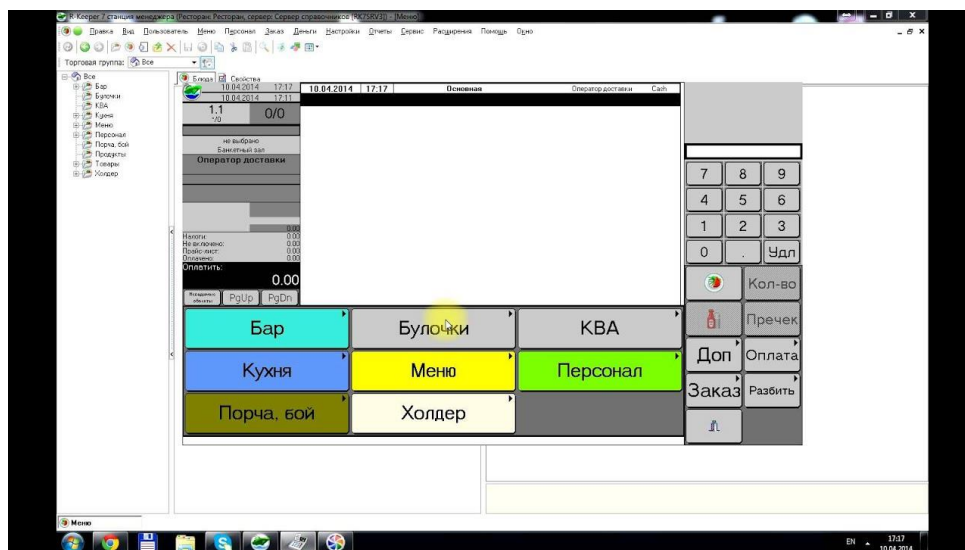


Рисунок 1.1.2 – пример решения «R-keeper»

На текущем рынке программных продуктов казахстанскому ресторатору, который решил автоматизировать свой бизнес, довольно сложно сделать правильный выбор.

С точки зрения рынка, ресторатор должен принимать решение, основываясь не только на стоимости программного обеспечения, но и на особенностях своей компании и перспективах ее развития.

Основная ошибка - неправильное понимание пользователем того, что они хотят получить после автоматизации. Система автоматизации не только предназначена для уменьшения ручной нагрузки, но и предназначена для анализа результатов этой работы. Помимо автоматизации текущих бизнес-процессов компании, ресторатор должен помнить, как выбранная система ограничит возможности развития бизнеса в будущем. При выборе системы важно не допустить, чтобы бизнес-возможности превышали возможности выбранной системы. Модернизация или обмен необходим. Главное правило: «автоматизируй сегодня, не забывая, что будет завтра».

Особым фактором при выборе системы автоматизации следует учитывать наличие поблизости сервисных служб. Например, компании имеют большую сеть партнеров в основных регионах Казахстана. Это позволяет вам быть гораздо ближе к клиенту не только как поставщик программного продукта и его поддержки, но и как поставщик различного торгового оборудования.

## 1.2 Анализ рынка HoReCa

Решения, предлагаемые лидерами, часто включают в себя новые инновационные функции, использование которых в данный момент кажется излишним для ресторатора, возможно, даже не логично.

Необходимость наличия в решении функции управления залом ресторана в некоторой степени понятна ресторатору. Чего нельзя сказать, например, о функции инвентаризации или управления затратами. Поскольку можно контролировать стоимость, ингредиенты имеют покупную цену поставщика, и управлять ею кажется бесполезным. Рестораторы считают, что зачем управлять нашими товарными запасами, ведь жили и до покупки системы автоматизации.

Но данные функции добавлены в решение совершенно не просто так. За каждым подобным бизнес-процессом лежит экономическая целесообразность его использования в реальной жизни.

Себестоимостью ингредиентов управлять сложно, уже есть сложившиеся взаимоотношения с поставщиками. Но стоит перенести эту концепцию на стоимость конечного блюда, потому что сразу понятно, зачем оно нужно. Замена различных ингредиентов их более дешевыми аналогами может снизить конечную стоимость блюда, увеличивая тем самым прибыль учреждения.

Одной из основных проблем, с которой сталкивается ресторатор при выборе системы автоматизации, является большое количество предлагаемых решений, которые больше похожи друг на друга [3]. Очень сложно понять существующие решения и какие из них наиболее подходят для ресторатора. Желание ресторатора понятно - минимизировать свои риски при установке выбранной системы автоматизации. Поэтому предложения от известных разработчиков «ресторанного программного обеспечения», с многолетним опытом автоматизации ресторанного рынка, кажутся более выгодными.

Проведя анализ, видим, что ресторатору не стоит пренебрегать многолетним опытом разработчиков программного обеспечения в ресторанной отрасли. Находясь на месте ресторатора, стоит выбрать более новый продукт и вот почему.

Более подробный анализ окончательной стоимости покажет, что эта стоимость включает в себя не только саму программу, но и учебные материалы для обучения различных категорий персонала, запуск, технологии быстрого внедрения продукта, ряд консультаций и поддержку пользователей и т.д.

Такие услуги не должны восприниматься ресторатором как навязанный пакет дополнительных услуг - это необходимость, продиктованная предыдущим опытом в области автоматизации и требованиями тех же коллег в предлагаемой системе. Использование дополнительных услуг - большой шаг к успешному внедрению системы автоматизации в компании.

### **1.3 Анализ существующих IT решений рынка общественного питания**

В последнее время рестораторы стали больше внимания уделять автоматизации своих предприятий. Этому требуют современные тенденции ресторанного рынка, растущая конкуренция, желание ресторатора сделать свой бизнес более прозрачным и легким в управлении.

Степень автоматизации в сфере общественного питания в Казахстане оставляет желать лучшего. Степень автоматизации предприятий общественного питания в Алматы и Астане достаточно высока, чего нельзя сказать об автоматизации аналогичных компаний в других регионах Казахстана.

Конечно, рынок автоматизации для предприятий общественного питания будет расти. Этот рост обусловлен не только развитием ресторанного бизнеса в регионах Казахстана. Существующие рестораны развиваются, добавляются кухонные фабрики, мастерские по производству полуфабрикатов и кондитерских изделий, услуги заказа и доставки для потребителей на дому. Кейтеринговые компании также расширяются.

Растет число сетей и растут существующие сети. Многие из тех, кто сейчас открывает первый ресторан, планируют в дальнейшем создание сети. Это неудивительно, поскольку уже признанный факт заключается в том, что сетевые концепции являются более прибыльными и более стабильными на рынке, чем отдельные рестораны[1].

Кроме того, пример успешной работы над концепцией гастрономического рынка сети иностранных брендов и растущей внутренней конкуренции может заставить казахстанских рестораторов объединиться и создать сеть ресторанов или холдингов.

Сложно определить основную тенденцию на рынке автоматических систем учета и контроля для ресторанного сектора. Базовый набор учетных функций, необходимых в системах, предлагаемых казахстанскими разработчиками, уже давно реализован. В разрабатываемой платформе, покажем несколько функций, на реализацию которых основные создатели «ресторанного софта» не обращают должного внимания.

Система автоматизации должна быть интегрированной системой управления. Функции автоматического выставления счетов и проверки недостаточны. Система должна содержать набор готовых бизнес-процессов, необходимых для эффективного и результативного управления бизнесом. Например, могут быть предоставлены следующие бизнес-процессы: управление запасами, производство, цены, продажи, наличные и т.д.

Система должна содержать набор готовых бизнес-процессов для работы в условиях распределенной сетевой структуры ресторанного холдинга. Управление сетевыми предприятиями существенно отличается от управления

одинокими. В качестве примера можно привести следующие «сетевые» бизнес-процессы:

- обеспечение контроля деятельности ресторанов на расстоянии;
- централизованная работа с поставщиками;
- контроль закупочных цен на местах;
- централизованное управление ценообразованием в целях обеспечения единого уровня сервиса для посетителей;
- сокращение количества персонала за счет централизации управления;
- анализ эффективности меню;
- открытие новых ресторанов по отлаженной схеме в короткие сроки.

Бизнес в сфере общественного питания, особенно в крупных городах, значительно вырос за последние несколько лет, а современные системы автоматизации оказывают неоценимую помощь в решении задач учета, управления и анализа деятельности предприятия [2]. При выборе системы для бизнеса необходимо учитывать, что на рынке представлено большое количество решений, каждое из которых имеет свои преимущества и недостатки. Представленный обзор для предприятий общественного питания еще раз подтверждает, что очень сложно выбрать наиболее эффективное решение. Можно оценить все плюсы и минусы различных программных продуктов только в практической деятельности. Но тот факт, что сеть общественного питания не может нормально функционировать без централизованного автоматического управления, не вызывает сомнений. И самая важная задача - научиться эффективно работать с выбранной системой автоматизации и максимально использовать ее.

#### **1.4 Постановка задачи**

Задача дипломного проекта заключается в создании автоматизированной системы заказов и продаж продуктов питания для сегмента HoReCa, открытия дополнительного канала продаж для поставщиков, а также минимизация рисков для поставщиков, ресторанов, отелей и сетей быстрого питания.

Разработанная платформа будет реализовываться в формате WEB – приложения. Платформа будет состоять из нескольких страниц. Первая страница это главная информационная страница, на остальных страницах будут размещены сами товары, форма входа, форма регистрации, страница добавления товаров, личный кабинет, категории и контактная информация.

Данная платформа позволит регистрироваться как поставщикам, так и ресторанам всего Казахстана. RestoFood позволит всем поставщикам выставлять свои товары, что откроет каждому поставщику дополнительный канал продаж, это значит что происходит минимизация рисков, так как у поставщика не будет происходить порча продуктов питания и соответственно он не будет терпеть убытков. Также платформа исключит риск распределения товара на складе поставщика, так как благодаря сайту он будет видеть весь

необходимый объем. Заказчикам (ресторанам) разработка позволит получить широкий ассортимент только качественных и свежих продуктов питания по низким ценам. Используя платформу заказчики минимизируют такие риски как:

- нехватка продуктов питания для готовки;
- испорченные и не свежие продукты;
- коррупционные схемы среди закупщиков ресторана;
- высокие цены;
- маленький ассортимент.

Чтобы реализовать данный проект необходимо изучить бизнес-процессы обеих сторон и осуществить следующие требования:

- анализ бизнес-процессов поставщиков и ресторанов;
- выбор оптимальных инструментов для разработки;
- разработка информационной системы (маркетплейс RestoFood);
- разработка веб-интерфейса для сайта.



## 2 Проектирование системы и обоснование выбора инструментов

На сегодняшний день процесс автоматизации общепита набирает обороты и ведется активная работа в данном направлении. Разработка таких информационных систем очень важна для анализа и минимизации рисков в сегменте HoReCa. В данном разделе описываются процессы проектирования и какие технологии были использованы при разработке дипломного проекта, их ключевые достоинства и недостатки, почему были выбраны именно эти технологии, и за какую часть они отвечают.

### 2.1 Проектирование закупочной системы общепита

Проектирование можно начать с диаграммы прецедентов, которая включает описание типичных взаимодействий между пользователями системы и самой системой и предоставление описания ее работы.

На рисунке 2.1.1 представлена диаграмма прецедентов для закупщика и ресторатора, то есть технология определения требований функционала к системе.

Система наделяет их следующими функциональными возможностями:

- регистрация;
- пользование личным кабинетом;
- просмотр товаров.

Всеми этими обязанностями обладает Ресторатор. Дополнительной возможностью является добавление товара.

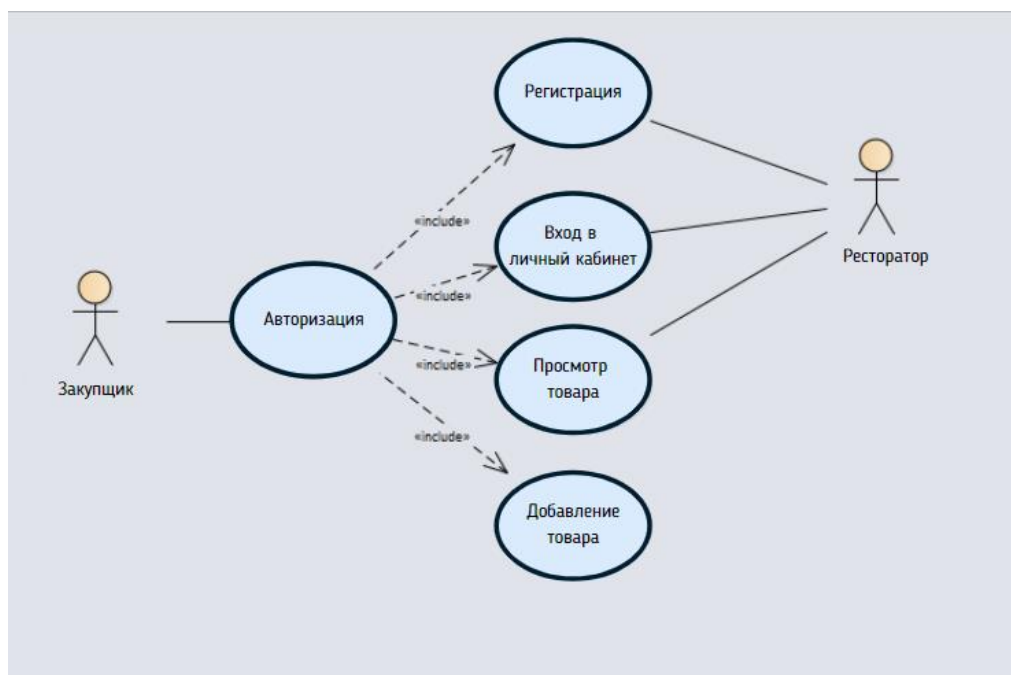


Рисунок 2.1.1 – диаграмма прецедентов для закупщика и ресторатора

На рисунке 2.1.2 представлена диаграмма прецедентов для пользователя. Система должна обеспечивать следующие функциональные возможности для Пользователя:

- вход для пользователя начинается с авторизации;
- просмотр категорий и товаров;
- просмотр карточки товара;
- заказ товара (при роли ресторатора);
- оформление заказа (при роли ресторатора);
- создание, удаление, редактирование товара (при роли поставщика);
- просмотр списков;
- добавление товаров в список желаний.

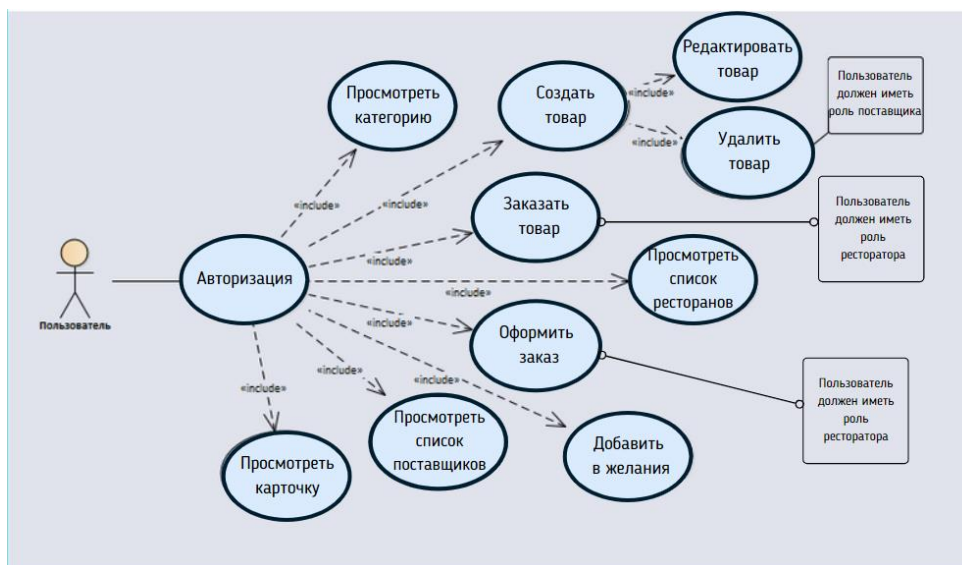


Рисунок 2.1.2 – Диаграмма прецедентов для пользователя

Серверная часть сайта должна содержать базу данных, а также программные функции для реализации регистрации в системе, создания и добавления товара, создания личного кабинета поставщика и ресторатора, восстановления в системе, редактирования товаров, формирования списка желаний, создания списков поставщиков и ресторанов и т.д. Опишем базу данных проекта. Структурная схема базы данных платформы Restofood представлена на рисунке 2.1.3.

На рисунке 2.1.3 можно увидеть схему базу данных которая была сделана при помощи таких инструментов как: phpMyAdmin и MySQL. В целом для создания всей платформы дополнительно были приобретены домен и виртуальный хостинг. Благодаря виртуальному хостингу получаем удобную систему для управления базой данных и всей серверной части платформы.

Для развертывания сайта на домене был взят адрес restofood.kz, который представлен на рисунке 2.1.4.

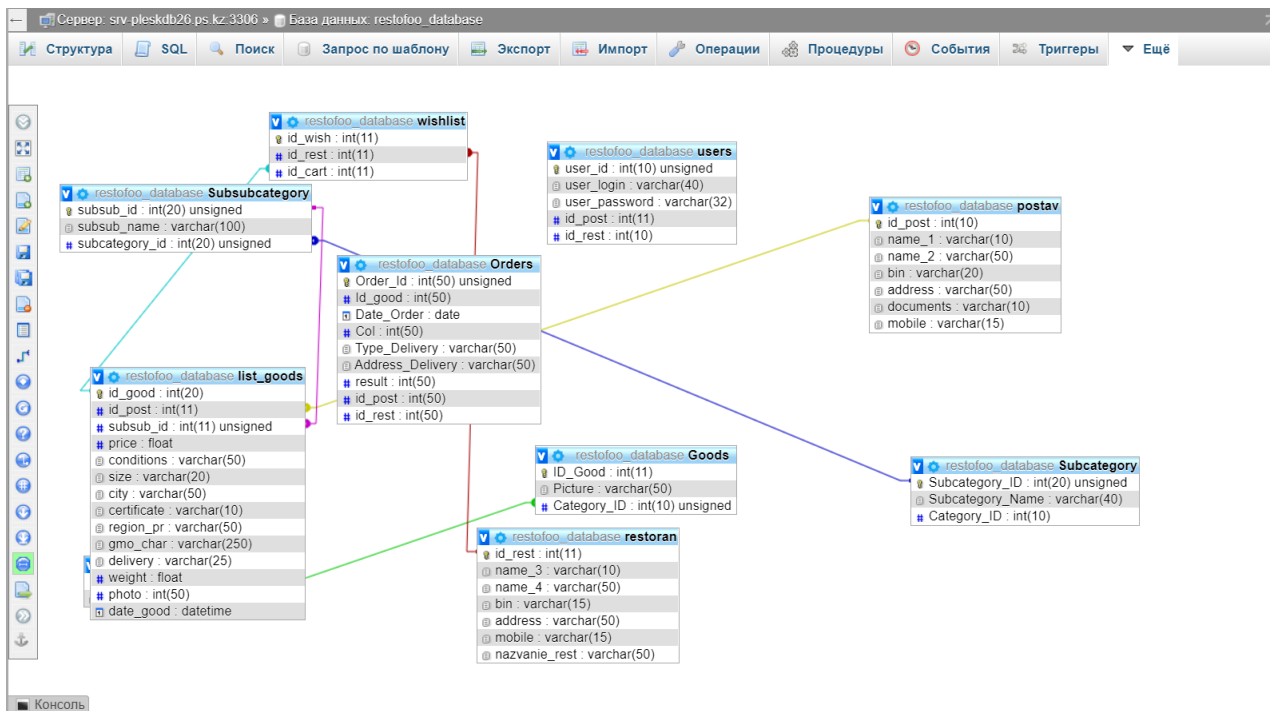


Рисунок 2.1.3 – Схема базы данных

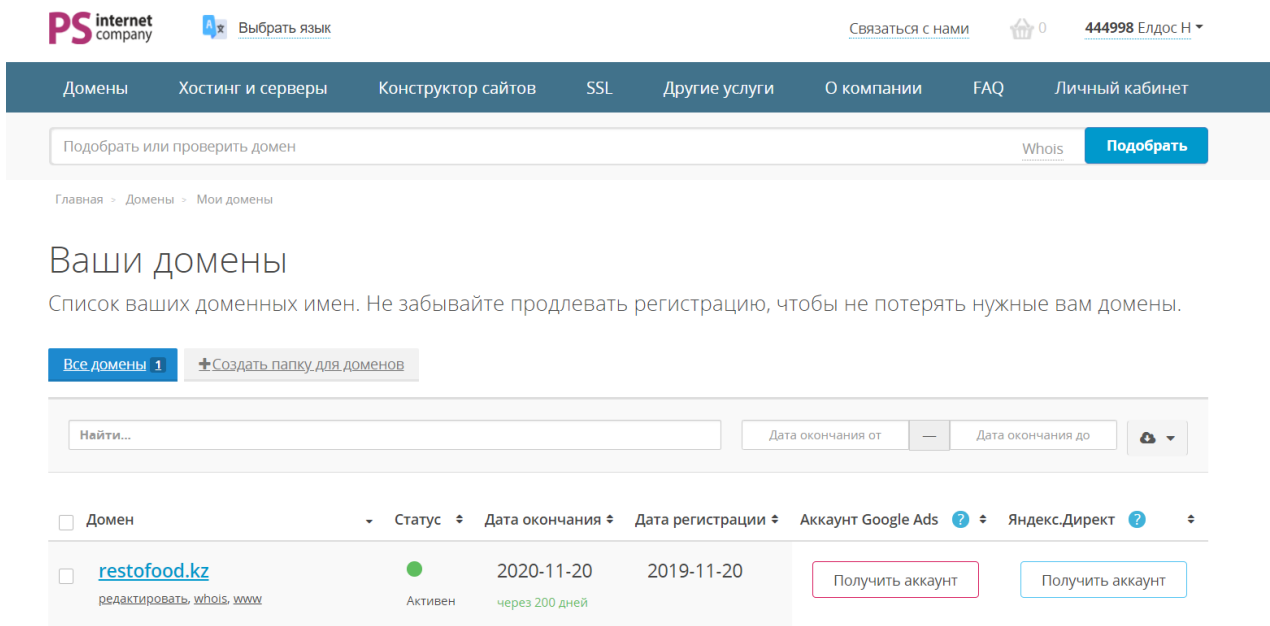


Рисунок 2.1.4 – Доменное имя нашей системы

Так как проект является действующим и тестируется ресторанами и поставщиками города Алматы, был приобретен домен, который представлен на рисунке 2.1.4.

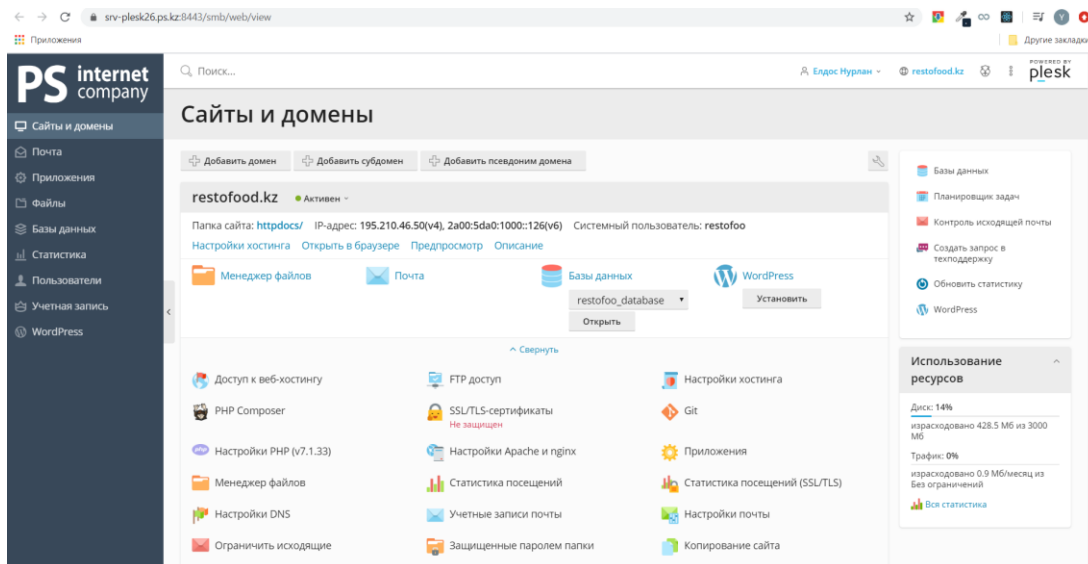


Рисунок 2.1.5 – Система управления серверной частью

На рисунке 2.1.5 представлена система управления серверной частью сайта, которая называется Plesk. Используя Plesk можно администрировать исходные файлы проекта, управлять базой данных, настраивать сетевые параметры, вести статистику.

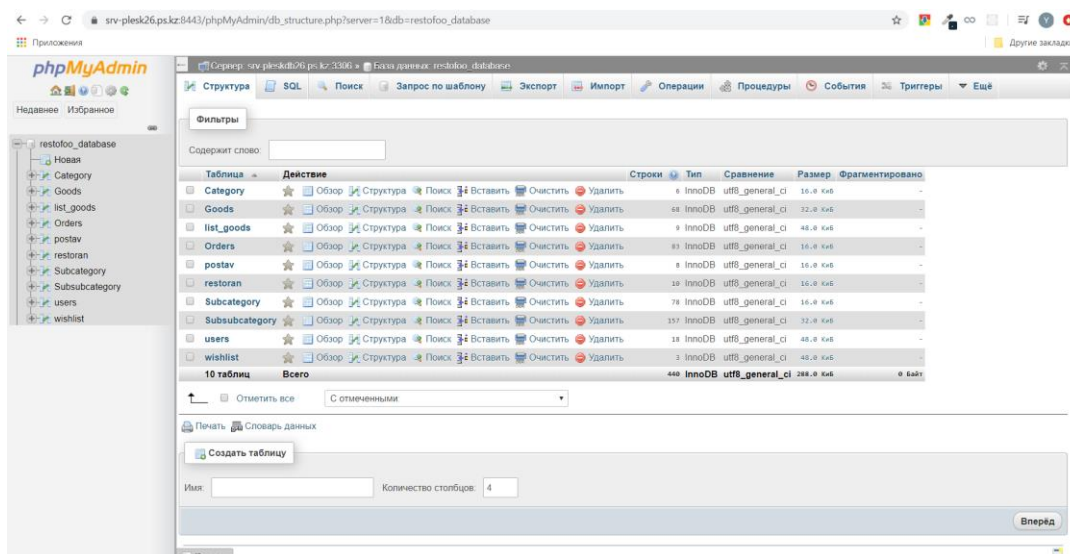


Рисунок 2.1.6 – Таблицы базы данных

На рисунке 2.1.6 представлены таблицы базы данных в количестве 10 штук, в них хранятся данные о пользователях (поставщики, рестораны), категориях продуктов, продуктах, подкатегориях, предпочтениях, заказах и т.д.

На рисунке 2.1.7 представлена внутренняя структура одной из таблиц базы данных. Это таблица с данными о поставщиках зарегистрированных на платформе, в ней указаны название, тип компании, БИН, адреса, документы и контактные данные.

```

284 CREATE TABLE `postav` (
285   `id_post` int(10) NOT NULL,
286   `name_1` varchar(10) NOT NULL,
287   `name_2` varchar(50) NOT NULL,
288   `bin` varchar(20) NOT NULL,
289   `address` varchar(50) NOT NULL,
290   `documents` varchar(10) NOT NULL,
291   `mobile` varchar(15) NOT NULL
292 ) ENGINE=InnoDB DEFAULT CHARSET=utf8;
293
294 --
295 -- Дамп данных таблицы `postav`
296 --
297
298 INSERT INTO `postav` (`id_post`, `name_1`, `name_2`, `bin`, `address`, `documents`, `mobile`) VALUES
299 (1, 'qqq', 'qqq', 'qqq', 'qqq', 'yes', 'qqq'),
300 (2, 'aa', 'aa', 'aa', 'aa', 'yes', 'aa'),
301 (3, 'fdsa', 'fdsa', 'fdsa', 'fdsa', 'yes', 'fdsa'),
302 (4, 'T00', 'Пинк', '90340204902', 'Журмангазы 112', 'yes', '87022891391'),
303 (5, 'TestPost', 'Тест', 'ТЕСТ', 'Курагова,2а, 6', 'yes', '8777777777'),
304 (6, 'T00', 'Дос', '48754156', 'Тоше би 285/5', 'yes', '+77052204203'),
305 (11, 'ИП', 'Good Delivery', '910613302316', 'Алматы', 'no', '+7 (775) 607 76'),
306 (12, 'ип', 'поставка', '77777777', 'муканова 87', 'yes', '+7 (444) 555 22');
307
308

```

Рисунок 2.1.7 – Внутренняя структура одной из таблиц

## 2.2 Выбор инструментов для разработки платформы

Для разработки предложенного проекта был использован стандарт последней версии-HTML5 (HyperText Markup Language). HTML5 - это обновление, сделанное в HTML из HTML4 (XHTML следует другой схеме нумерации версий). Он использует те же основные правила, что и HTML 4, но добавляет некоторые новые теги и атрибуты, которые позволяют улучшить семантику и динамические элементы, активируемые с помощью JavaScript.

Впервые разработанный Тимом Бернерсом-Ли в 1990 году, HTML-это сокращение от языка разметки гипертекста (HyperText Markup Language). HTML используется для создания электронных документов (называемых страницами), которые отображаются во Всемирной паутине [4]. Каждая страница содержит ряд соединений с другими страницами, называемых гиперссылками. Каждая веб-страница, которую вы видите в Интернете, написана с использованием той или иной версии HTML-кода.

HTML-код обеспечивает правильное форматирование текста и изображений для вашего интернет-браузера. Без HTML браузер не будет знать, как отображать текст в виде элементов или загружать изображения или другие элементы. HTML также предоставляет базовую структуру страницы, на которую накладываются Каскадные Таблицы Стилей (CSS - Cascading Style Sheets) для изменения ее внешнего вида. Можно представить себе HTML как кости (структуру) веб-страницы, а CSS-как ее оболочку (внешний вид).

CSS - это язык таблиц стилей, который позволяет изменять внешний вид языка разметки. Например, HTML используется для создания базового макета веб-страницы, например этого абзаца текста. CSS используется для определения шрифта, размера шрифта, веса шрифта, его положения и других визуальных настроек [12].

Сокращенно от cascading style sheets, CSS-это язык, используемый для описания многоцветных стилей представления документов, написанных на языке разметки. Его концепция была создана Hakon Wium Lie в 1994 году. В декабре 1996 года CSS был внесен в спецификацию W3C и сегодня позволяет веб-разработчикам изменять макет и внешний вид своих веб-страниц. Например, CSS может использоваться для изменения шрифта, используемого в определенном элементе HTML, а также его размера и цвета. Один CSS-файл может быть связан с несколькими страницами, что позволяет разработчику изменять внешний вид всех страниц одновременно.

Перед началом работы был изучен язык программирования PHP, шаблоны и методики программирования. Выяснилось что PHP в общем-то без особых проблем позволяет реализовывать те или иные паттерны разработки. То есть в PHP можно писать правильный код, который будет не сильно отличаться от того, что получаем на Python/Java.

Проще говоря, PHP-это аббревиатура от слова «hypertext preprocessor». Это наиболее часто используемый серверный скриптовый язык, который используется преимущественно для веб-разработки. PHP является наиболее широко и широко используемым серверным сайтом в интернете, поскольку он, как известно, управляет или охватывает почти 82 % всего интернета сегодня. Помимо его использования в веб-разработке, PHP также используется в качестве языка программирования общего назначения. Компьютерный язык PHP был впервые использован в истории в 1995 году и, как известно, был разработан за год до этого великим компьютерным ученым Расмусом Лердфом. Первоначально PHP обозначал «личную домашнюю страницу», которая в последующие годы была заменена на «гипертекстовый препроцессор» [25].

Помимо разработки веб-сайтов, PHP используется с целью разработки веб-программ. Одна очень уникальная и контрастная особенность компьютерного языка PHP заключается в том, что его скрипты могут быть интерпретированы только на серверах, на которых уже установлен PHP. Еще одна контрастная особенность языка PHP заключается в том, что компьютеры клиентов, которые фактически получают доступ к веб-сайту, разработанному с использованием PHP, не требуют установки PHP на своем компьютере, им просто требуется веб-браузер для доступа к нему. PHP совместим в основном с Windows, а также с Unix-подобными операционными системами.

В нашем проекте применен JavaScript для разработки модальных окон, таких как: восстановление пароля, оформление заказа, редактирование товара. Он был выбран, так как скрипты не требуют перезагрузки страницы, что упрощает для пользователей работу с системой. JavaScript - это, как известно клиентский скрипт. Большинство веб-приложений, таких как поисковая система, работают благодаря взаимодействию между устройством пользователя (например, компьютером, телефоном или планшетом) и удаленным сервером. Программное обеспечение на удаленном сервере отправляет информацию клиенту (т. е. машине пользователя), а программное

обеспечение на стороне клиента считывает информацию и выводит веб-страницу на экран.

JavaScript - это язык программирования, используемый в основном веб-браузерами для создания динамического и интерактивного контента для пользователя. Большинство функций и приложений, которые делают Интернет незаменимым в современной жизни, закодированы в той или иной форме JavaScript.

Самые ранние воплощения JavaScript были разработаны в конце 1990-х годов для веб-браузера Netscape Navigator[22]. В то время веб-страницы были статичными, предлагая мало взаимодействия с пользователем, кроме щелчка по ссылкам и загрузки новых страниц. Впервые JavaScript включил анимацию, адаптивное содержимое и проверку формы на странице.

В течение многих лет JavaScript функционировал только в ограниченном количестве браузеров. Explorer от Microsoft, крупнейший базовый браузер стал поддерживать JavaScript намного позже. Вместо этого Microsoft создала свой собственный клиентский скрипт под названием JScript. В первые дни веб-разработки программисты, которые хотели создавать динамические веб-сайты, часто были вынуждены выбирать одно семейство браузеров вместо другого. Это было далеко не идеально, поскольку делало Интернет менее общедоступным.

Для разработки базы данных была выбрана СУБД MySQL, так как при приобретении виртуального сервера, MySQL является системой управления реляционными базами данных по умолчанию и ввиду того, что наша система требует особых условий к скорости работы БД, масштабируемости и др., то есть можем использовать MySQL ввиду её высокой популярности, относительно высокой скорости работы, наличию большого количества документации и большого сообщества разработчиков.

MySQL-это поддерживаемая Oracle система управления реляционными базами данных с открытым исходным кодом, основанная на языке структурированных запросов (SQL) [11]. MySQL работает практически на всех платформах, включая Linux, UNIX и Windows. Хотя он может быть использован в широком спектре приложений. MySQL основан на модели клиент-сервер. Ядром MySQL является сервер MySQL, который обрабатывает все инструкции базы данных (или команды). Сервер MySQL доступен как отдельная программа для использования в сетевой среде клиент-сервер и как библиотека, которая может быть встроена (или связана) в отдельные приложения.

MySQL работает вместе с несколькими служебными программами, которые поддерживают Администрирование баз данных MySQL. Команды отправляются на сервер MySQL через клиент MySQL, который установлен на компьютере. MySQL изначально был разработан для быстрой обработки больших баз данных. Хотя MySQL обычно устанавливается только на одной машине, он способен отправлять базу данных в несколько местоположений, поскольку пользователи могут получить к ней доступ через различные

клиентские интерфейсы MySQL. Эти интерфейсы отправляют инструкции SQL на сервер, а затем отображают результаты. MySQL позволяет хранить данные и получать к ним доступ через несколько механизмов хранения, включая InnoDB, CSV и NDB. MySQL также способен реплицировать данные и разбивать таблицы на разделы для повышения производительности и долговечности. Пользователи MySQL не обязаны изучать новые команды; они могут получить доступ к своим данным с помощью стандартных команд SQL.

Для работоспособности и использования заказчиками платформы RestoFood было приобретено доменное имя и виртуальный хостинг с панелью управления Plesk, благодаря которой можно управлять базой данных и всеми файлами платформы.

Parallels Plesk Panel или просто Plesk вкратце известен как одна из самых многофункциональных панелей управления на планете. Это выходит далеко за рамки возможностей обычных веб-инструментов для управления серверными ресурсами, необходимыми для существования ваших сайтов. Plesk - это полноценная многоуровневая панель управления хостинговой платформой, предоставляющая администратору сервера широкий спектр возможностей графического интерфейса для настройки и обслуживания всего хостингового бизнеса, обслуживающего конечного клиента. Он так же хорош для настройки общих хостинговых решений, как и для настройки виртуальных и выделенных серверов. Операционные системы, поддерживающие Plesk, включают Fedora, Red Hat, Debian, FreeBSD, SuSE, CentOS, Ubuntu и Windows.

Панель управления Plesk может похвастаться простым в навигации интерфейсом автоматизации веб-хостинга. Навигация по его многочисленным разделам была максимально облегчена, так что вы могли бы быстро найти необходимую вам функциональность в любой момент времени. Панель управления запоминает все наиболее часто используемые функциональные возможности и приносит их вам в любое время, когда вы получаете доступ к ее интерфейсу.



### 3 Разработка программного продукта

Приложение клиент-сервер - это приложение, которое имеет две части: клиент и сервер. Эти приложения обычно отображаются через веб-браузер и взаимодействуют с пользователем.

Клиент-это часть приложения, которая отображается пользователю, выполняется в веб-браузере и визуально взаимодействует с пользователем. На этой странице работают языки разметки, стиля и программирования, такие как HTML, CSS и JavaScript.

Серверная часть приложения не имеет собственного визуального представления и взаимодействует с пользователем через веб-браузер. Название этой части происходит от того, что все действия выполняются на сервере — специальном компьютере, который может находиться за тысячи километров от браузера или рядом с машиной. Сервер обычно размещает базу данных и использует такие языки, как Java, PHP, C# и т. д. Это приложение разработано на языке программирования PHP.

Платформа будет состоять из нескольких страниц. Первая страница это главная информационная страница, на остальных страницах будут размещены сами товары, форма входа, форма регистрации, страница добавления товаров, личный кабинет, категории и контактная информация.

Прежде чем разрабатывать клиентскую часть нужно продумать бизнес-процессы всех сторон и начать разработку.

#### 3.1 Системные требования

Рекомендуемые системные требования:

- операционная система: Windows 7/8/10 — 64-bit.
- процессор (CPU): Intel Core i3.
- ОЗУ (RAM): 1 ГБ (или больше).
- свободное место на жёстком диске: ~35 МВ.

Минимальные системные требования:

- операционная система: Windows 7/8/10 — 64-bit.
- процессор (CPU): Intel Pentium.
- ОЗУ (RAM): 512 МВ (или больше).
- свободное место на жёстком диске: ~35 МВ.

#### 3.2 Описание разработанных функций

Для написания программной части был применен язык разметки HTML, PHP, JavaScript, а также для придания стилей был применен CSS. Для развертывания информационной системы был приобретен домен и виртуальный хостинг, а также применена система управления файлами сайтами и базой данных.

Вначале юзер проходит регистрацию через соответствующую форму. Далее после заполнения полей данные отправляются для обработки на сервер, где попадают в таблицу БД. При следующем посещении на web-платформу наш пользователь вводит указанные при регистрации логин и пароль. Их правильность проверяется путем выборки данных из таблицы. Функционал регистрации представлен фрагментом на рисунке 3.2.6.

```
114         });
115
116     </script>
117     <script type="text/javascript">
118 window.onload = function() {
119 //document.getElementById('restoname').style.display = 'none';
120 document.getElementById('radiodoc').style.display = 'none';}
121 function yesnoCheck() {
122     if (document.getElementById("role2").checked==true) {
123         document.getElementById("restoname").style.display = 'none';
124         document.getElementById("radiodoc").style.display = 'block';
125         document.getElementById("namerest").required = false;
126     } else if (document.getElementById("role1").checked==true) {
127         document.getElementById("restoname").style.display = 'block';
128         document.getElementById("radiodoc").style.display = 'none';
129         document.getElementById("namerest").required = true;
130     }
131 }
```

Рисунок 3.2.1 – Фрагмент кода для регистрации

Важной частью нашего проекта является функция авторизации на портале, так как уже тут происходит разделение по ролям, то есть на личные кабинеты поставщика и ресторатора. Для авторизации пользователя соединяемся с БД, затем вытаскиваем из БД запись, у которой логин равняется введенному. Далее сравниваем пароли, генерируем случайное число и шифруем его, затем записываем в БД новый хеш авторизации и IP, ставим cookie, переадресовываем браузер на страницу проверки нашего скрипта. Часть скрипта для авторизации пользователя представлена на рисунке 3.3.1.

```

login.php ×
9 // Соединяемся с БД
10 $link=mysqli_connect("srv-pleskdb26.ps.kz:3306", "resto_admin", "123qwerty123", "restofoo_database");
11 if(!isset($_SESSION['id'])){
12 if(isset($_POST['submit']))
13 {
14
15     $err = [];
16
17     if(!filter_var($_POST['login'], FILTER_VALIDATE_EMAIL))
18     {
19         $err[] = "Email указан неверно!";
20     }
21
22     $query = mysqli_query($link, "SELECT user_id FROM users WHERE user_login='".mysqli_real_escape_string($link, $_POST['login'])."'");
23     if(mysqli_num_rows($query) > 0)
24     {
25         $query = mysqli_query($link, "SELECT user_id, user_password, id_post, id_rest FROM users WHERE
user_login='".mysqli_real_escape_string($link, $_POST['login'])."' LIMIT 1");
26         // Вытаскиваем из БД запись, у которой логин равняется введенному
27         $data = mysqli_fetch_assoc($query);
28

```

Рисунок 3.2.2 – Скрипт для авторизации пользователя

Как видим на рисунке 3.2.2 пользователь заходит на страницу авторизации, чтобы попасть в свой личный кабинет и начать использовать сервис. Но бывает так, что пользователь может забыть свой пароль, поэтому в сервисе предоставлено восстановление пароля. Функция по восстановлению пароля представлена на рисунке 3.2.3.

```

245         </script>";
246         mail("$address", "Восстановление пароля", "e-mail: ".$data['user_login'] . "\r\n". "Пароль: ".$data['user_password'] .
"\r\n", "From: vvv@restofood.kz \r\n");
247
248     }
249
250     else
251     {
252         echo "<script>
253         $('#myModalF').modal('show');
254         </script>";
255     }
256 }
257 ?>
258 <!-- HTML-код модального окна-->
259 <div id="myModal1" class="modal fade">
260
261     <div class="modal-dialog" role="document">
262         <div class="modal-content">
263             <div class="modal-header">
264                 <button type="button" class="close" data-dismiss="modal" aria-label="закреть">

```

Рисунок 3.2.3 – Скрипт для восстановления пароля

Для разделения ролей между поставщиками и ресторатором используется личный кабинет, фрагмент кода которого показан на рисунке 3.2.4.

Данный скрипт служит для разделения ролей, вывода данных поставщика или ресторана и отображения история заказов для ведения статистики. Содержимое файла истории заказов показано на рисунке 3.2.5.

Также создав личный кабинет позволено пользователям системы следить за своими расходами и статистикой что минимизировало риски. Также поставщикам открыта возможность просматривать свои товары, а также при необходимости редактировать и удалять их.

```

67 <?php
68 session_start();
69 $type=$_SESSION['typeaccount'];
70 $link=mysqli_connect("srv-pleskdb26.ps.kz:3306", "resto_admin", "123qwerty123", "restofoo_database");
71 $sql_select = "SELECT * FROM restoran WHERE id_rest='".mysqli_real_escape_string($link, $_SESSION['typeaccount1'])."'";
72 $result = mysqli_query($link,$sql_select);
73 $row = mysqli_fetch_array($result);
74
75 $sql_select1 = "SELECT * FROM postav WHERE id_post='".mysqli_real_escape_string($link, $_SESSION['typeaccount1'])."'";
76 $result1 = mysqli_query($link,$sql_select1);
77 $row1 = mysqli_fetch_array($result1);
78
79
80 if($type=='restoran') {
81     printf("<a href='\"index4.php\"' class='\"d-block\">\" . $row['name_3'] . \"\" . $row['name_4'] . \"</a> \");
82 }
83 elseif ($type=='postav')
84 {
85     printf("<a href='\"index4.php\"' class='\"d-block\">\" . $row1['name_1'] . \"\" . $row1['name_2'] . \"</a> \");
86 }

```

Рисунок 3.2.4 – Фрагмент кода личного кабинета

```

148 //История заказов ресторана
149 $sql_select0 = "SELECT * FROM users WHERE user_id='".mysqli_real_escape_string($link, $_SESSION['id'])."'";
150 $result0 = mysqli_query($link,$sql_select0);
151 $row0 = mysqli_fetch_array($result0);
152 $sql_select2 = "SELECT * FROM Orders WHERE id_rest='".mysqli_real_escape_string($link, $row0['id_rest'])."'";
153 $result2 = mysqli_query($link,$sql_select2);
154 $row2 = mysqli_fetch_array($result2);
155 //История заказов поставщика
156
157 $sql_selectP = "SELECT * FROM users WHERE user_id='". mysqli_real_escape_string($link, $_SESSION['id']) . "'";
158 $resultP = mysqli_query($link,$sql_selectP);
159 $rowP = mysqli_fetch_array($resultP);
160 $sql_select6 = "SELECT * FROM Orders WHERE id_post='". mysqli_real_escape_string($link, $rowP['id_post']) . "'";
161 $result6 = mysqli_query($link,$sql_select6);
162 $row6 = mysqli_fetch_array($result6);
163
164
165
166 if($type=='restoran') {
167     printf("

```

Рисунок 3.2.5 – Содержимое файла истории заказов

На рисунке 3.2.6 представлена часть кода, где поставщики могут смотреть свои товары, редактировать и удалять их.

```

160 if ($type=='postav') {
161     echo ' <h1 class="m-0 text-dark">Мои товары</h1>';
162     echo "<form action='\"\"' method='\"post\">";
163     while ($row8 = mysqli_fetch_array($result8)) {
164         $sql_select9 = "SELECT * FROM Subsubcategory WHERE subsub_id='". mysqli_real_escape_string($link, $row8['subsub_id']) . "'";
165         $result9 = mysqli_query($link, $sql_select9);
166         $row9 = mysqli_fetch_array($result9);
167
168         $sql_select5 = "SELECT * FROM Goods WHERE ID_Good='". mysqli_real_escape_string($link, $row8['photo']) . "'";
169         $result5 = mysqli_query($link, $sql_select5);
170         $row5 = mysqli_fetch_array($result5);
171         printf("
172 <hr> <div class='\"info-box mb-3 bg-warning\">
173 <img src='\"../\".$row5['Picture'].\"' width='100' height='100'>
174
175 <div style='padding-bottom:5px' class='\"info-box-content\">
176 <input name='idgood' value='\" . $row8['id_good'] . \"' hidden>
177 <span style=' display: block; float: left; width: 200px; height: 15px;'> Название:</span><b>\" . $row9['subsub_name'] . \" </b>
<br>
178 <span style='display: block; float: left; width: 200px; height: 15px;'>Вес:</span> <b>\" . $row8['weight'] . \"</b> <br>
179 <span style='display: block; float: left; width: 200px; height: 15px;'>Цена:</span> <b>\" . $row8['price'] . \"</b> <br>

```

Рисунок 3.2.6 – Функция для просмотра и удаления товаров

После регистрации поставщика можно загрузить свои товары. Функция загрузки товара на сервер и его отображения в системе представлена на рисунке 3.2.7.

```
201         $cat = $_POST['categoria'];
202
203         //echo $cat;
204         echo '<select name = "categoria2" onChange="this.form.submit();">';
205
206         if($_POST['categoria2']!= ''){
207             $cat2 = $_POST['categoria2'];
208             $sqli = "SELECT * FROM Subcategory WHERE Subcategory_ID=$cat2";
209             $result = mysqli_query($link, $sqli);
210             $data = mysqli_fetch_assoc($result);
211             echo '<option value='.$cat2.'>'.$data['Subcategory_Name'].'</option>';
212         }
213         echo '<option ></option>';
214
215         $sqli = "SELECT * FROM Subcategory WHERE Category_ID=$cat";
216         $result = mysqli_query($link, $sqli);
217         while ($row = mysqli_fetch_array($result)) {
218             echo '<option value='.$row['Subcategory_ID'].'>'.$row['Subcategory_Name'].'</option>';
219         }
220     }
221 }
```

Рисунок 3.2.7 – Функция добавления товаров

После добавления товаров поставщиками, товары отображаются на отдельной странице. Чтобы пользователям было легче ориентироваться по товарам, была организована функция навигации, где присутствуют поиск, фильтр и сортировка по выбранным товарам. Часть функции навигации представлена на рисунке 3.2.8.

```
4 $link=mysqli_connect("srv-pleskdb26.ps.kz:3306", "resto_admin", "123qwerty123", "restofoo_database");
5
6 function link_bar($page, $pages_count)
7 {
8     echo '<ul class="pagination pagination-sm">';
9     for ($j = 1; $j <= $pages_count; $j++)
10    {
11        // Вывод ссылки
12        if ($j == $page) {
13            echo '<li><a style="color: #808000;" ><b>'.$j.</b></a></li> ';
14        } else {
15            echo '<li><a style="color: #808000;" href='.$_SERVER['php_self'].'?
16            page='.$j.'&categoria='.$_GET['categoria'].'&categoria2='.$_GET['categoria2'].'&categoria3='.$_GET['categoria3'].'&search_field='.$_GET['search_fie
17            sort='.$_GET['sort'].'>'.$j.</a></li> ';
18        }
19        // Выводим разделитель после ссылки, кроме последней
20        // например, вставить "|" между ссылками
21        /*if ($j != $pages_count) echo '|';*/
22    }
23    echo '</ul>';
24 }
```

Рисунок 3.2.8 – Часть функции навигации товаров

Чтобы увидеть товар формируется карточка товара, данные которые берутся из данных введенных поставщиком. В сформированной карточке товара отображается фотообложка, цена, месторасположение, тип доставки, наличие сертификатов, контакты, характеристики и еще есть дополнительная функция отображения похожего товара у других поставщиков. Формирование карточки товара отображено на рисунке 3.2.9.

```

360 <div class="modal-header">
361 <button type="button" class="close" data-dismiss="modal" aria-label="Заккрыть">
362 <span aria-hidden="true">x</span>
363 </button>
364 <h4 class="modal-title" id="myModalLabel2">Изменение товара</h4>
365 </div>
366 <div class="modal-body">
367
368 <form action="" method="post" onload="f();">
369 <div class="form-group row">
370 <label class="col-sm-4 col-form-label">Цена (тг.): </label>
371 <div class="col-sm-10">
372 <?php
373 $sqli = "SELECT list_goods.price FROM list_goods,Subsubcategory,Subcategory,Category,Goods WHERE
list_goods.id_good=".$GET['idgood']." AND list_goods.subsub_id=Subsubcategory.subsub_id AND Subsubcategory.Subcategory_ID=Subcategory.Subcategory_ID AND
Subcategory.Category_ID=Category.Category_ID AND Category.Category_ID=Goods.Category_ID ";
374 $result = mysqli_query($link, $sqli);
375 $data = mysqli_fetch_assoc($result);
376 $price=$data['price'];
377 echo <span> </span><input name="priceE" class="form-control" value="'. $price.' ">";
378

```

Рисунок 3.2.9 – Формирование карточки товара

Чтобы открыть дополнительный канал для продаж поставщикам, реализовали функцию просмотра всех ресторанов для поставщиков, а ресторанам просмотр всех поставщиков. Продавцы дополнительно могут просматривать желаемые товары рестораторов. Функция реализации списков и просмотра списка желаемого представлена на рисунке 3.2.10.

```

107 <p>Номер телефона: " . $row1['mobile'] . "</p><p>Адрес: " . $row1['address'] . "</p><p>БИН: " . $row1['bin'] . "</p><div class=\"clearfix\"></div>
<br/>";
108 );
109 echo <script>
110 $("#type_z").text('\Поставщики\');
111 </script>;
112 } while ($row1 = mysqli_fetch_array($result1));
113 }
114 }
115 elseif ($type=='postav')
116 {
117 do {
118 printf("<p>Название ресторана: " . $row['nazvanie_rest'] . "</p>
119 <p>Номер телефона: " . $row['mobile'] . "</p><p>Адрес: " . $row['address'] . "</p><p>БИН: " . $row['bin'] . "</p>";
120 );
121 echo <p><a class="account" href="wishlist1.php?idrest='.$row['id_rest'].' ">Посмотреть список желаемого ресторана</a><br><br><br><div
class="clearfix\"></div><br></p>;
122 echo <script>
123 $("#type_z").text('\Рестораны\');
124 </script>;
125 } while ($row = mysqli_fetch_array($result));

```

Рисунок 3.2.10 – Функция реализации списков

В списках также можно добавить товар в список желаемого, ниже будет описана эта функция.

Чтобы поставщики могли облегчить и автоматизировать свои процессы, для них была создана функция добавления товара в список желаемого. После того как поставщик создаст продукт и система сформирует карточку товара, закупщики ресторана могут добавить желаемый товар в список. Уже после продавцы смотрят какой продукт необходим тому или иному ресторану и могут предложить свою продукцию заинтересованным местам общепита. Также рестораны могут редактировать свой список пожеланий. Данный процесс изображен на рисунке 3.2.11.

```

172
173 // Если номер страницы оказался больше количества страниц
174 if ($page > $pages_count) $page = $pages_count;
175 $start_pos = ($page - 1) * $perpage; // Начальная позиция, для запроса к БД
176 $result = mysqli_query($link, $sqli.' LIMIT '.$start_pos.', '.$perpage);
177 if (empty($result))
178 {
179     echo "В списке желаний отсутствуют товары";
180 }
181 else{$i=0;
182     while ($row = mysqli_fetch_array($result)) {
183         echo '<div class="post-ad-form">
184             <a href="single.php?idgood='.$row['id_good'].'&categoria='.$row['subsub_id'].'&post='.$row['id_post'].'">
185                 <li>
186                     
187                     <section class="list-left">
188                         <h5 class="title">'.$row['subsub_name'].'</h5>
189                         <span class="adprice">'.$row['price'].'<tr.</span>
190                     <p
class="catpath">'.$row['Category_Name'].'/'.$row['Subcategory_Name'].'/'.$row['subsub_name'].'</p>
191                 </section>

```

Рисунок 3.2.11 – Функция добавления пожеланий

```

446     <label for="name">Доставка:</label>
447     <?php
448     $sqli = "SELECT list_goods.delivery FROM list_goods,Subsubcategory,Subcategory,Category,Goods WHERE
list_goods.id_good=".$GET['idgood']." AND list_goods.subsub_id=Subsubcategory.subsub_id AND Subsubcategory.Subcategory_ID=Subcategory.Subcategory_ID AND
Subcategory.Category_ID=Category.Category_ID AND Category.Category_ID=Goods.Category_ID ";
449     $result = mysqli_query($link, $sqli);
450     $data = mysqli_fetch_assoc($result);
451     $delivery=$data['delivery'];
452     echo '<span>'.$delivery.'</span><input name="delivery" hidden value="'.$delivery.'"> ';
453     ?>
454 </div>
455 <div class="form-group">
456     <label for="message">Адрес:</label>
457     <textarea class="form-control" name="message" rows="1"></textarea>
458 </div>
459 <script>
460     function f(){
461         var n1=document.getElementById('kol').value;
462         var n2=document.getElementById('priceG').innerHTML;
463         document.getElementById('total').textContent=parseInt(n1)*parseInt(n2);
464         document.getElementById('total1').value=parseInt(n1)*parseInt(n2);

```

Рисунок 3.2.12 – Функция оформления заказа

Для удобства оформления заказа ресторанам была разработана форма оформления заказа с использованием PHP и JavaScript скриптов. Всплывает модальное окно, в котором закупщики указывают цену, количество и адрес. Далее нажимают кнопку заказать и уже после заказ попадает в историю заказов, которая находится в личном кабинете. Процесс оформления заказа представлен на рисунке 3.2.12.

```

269
270
271     </ul>
272     <script type="text/javascript">
273         $(window).load(function() {
274             $("#flexiselDemo3").flexisel({
275                 visibleItems:1,
276                 animationSpeed: 500,
277                 autoPlay: true,
278                 autoPlaySpeed: 8000,
279                 pauseOnHover: true,
280                 enableResponsiveBreakpoints: true,
281                 responsiveBreakpoints: {
282                     portrait: {
283                         changePoint:480,
284                         visibleItems:1
285                     },
286                     landscape: {
287                         changePoint:640,
288                         visibleItems:1
289                     },
290                     tablet: {

```

Рисунок 3.2.13 – Код слайдера товаров

Также все новые добавленные поставщиками товары отображаются при загрузке главной страницы. В слайдере появляются новые и актуальные предложения для ресторанов нашего города. Для разработки слайдера с товарами был использован скрипт JavaScript. Код для разработки слайдера можно рассмотреть на рисунке 3.2.13.

```

211     <div class="side-bar col-md-3">
212         <div class="search-hotel">
213             <h3 class="sear-head">Поиск</h3>
214             <form>
215                 <?php
216                 if(!empty($_GET['search_field'])){echo '<input type="text" id = "search_fieldid" name="search_field"
value="'. $_GET['search_field'].'" required="" style="width: 220px">';}
217                 else{echo '<input type="text" id = "search_fieldid" name="search_field" value="" required="" style="width: 220px">';}
218                 <input type="submit" name="search_button" value="Найти" style="width: 220px">
219             </form>
220         </div>
221         <div class="range">
222         <h3 class="sear-head">Выбрать категорию</h3>
223         <select id = "categoria" name = "categoria" onChange="search_fieldidCH();this.form.submit();" style="width: 220px"><?php
224             if($_GET['categoria']!='){
225                 echo '<option value="'. $_GET['categoria'].'">'. $_GET['categoria']. '</option>';
226             }
227             ?>
228
229         <option value="">Выберите категорию</option>

```

Рисунок 3.2.14 – Функция поиска товаров

Для удобства поиска товаров как поставщикам, так и рестораторам была реализована функция поиска. Была сверстана секция с поисковиком и сортировкой, программная часть была реализована с помощью скриптов PHP. Реализация поисковика показана на рисунке 3.2.14.



```

221 <div class="range">
222 <h3 class="sear-head">Выбрать категорию</h3>
223 <select id = "categoria" name = "categoria" onChange="search_fieldidCH();this.form.submit();" style="width: 220px"><?php
224     if($_GET['categoria']!='){
225         echo '<option value="">'.$_GET['categoria'].'</option>';
226     }
227     ?>
228
229     <option value="">Выберите категорию</option>
230     <option value = "Мясо">Мясо</option>
231     <option value = "Овощи">Овощи</option>
232     <option value = "Фрукты">Фрукты</option>
233     <option value = "Зелень">Зелень</option>
234     <option value = "Грибы">Грибы</option>
235     <option value = "Молочные продукты">Молочные продукты</option>
236 </select>
237 </div>
238
239
240 <div class="range">
241 <h3 class="sear-head">Выбрать категорию</h3>

```

Рисунок 3.2.15 – Сортировка по категориям

Как описано ранее также была реализована сортировка продуктов по категориям, подкатегориям и по цене (возрастание и убывание). Процессы сортировки по категориям указаны на рисунке 3.2.15, сортировки по цене указаны на рисунке 3.2.16.

```

315 <div class="sort">
316 <div class="sort-by">
317 <label>Сортировка по: </label>
318 <select name="sort" onChange="this.form.submit();">
319 <?php
320
321     if($_GET['sort']=== "DESC"){
322         echo '<option value="DESC">Цена: По убыванию</option>';
323     }
324     else if($_GET['sort']=== " ") {echo '<option value="">Цена: По возрастанию</option>';}
325     ?>
326
327     <option value="">Цена: По возрастанию</option>
328     <option value="DESC">Цена: По убыванию</option>
329 </select>

```

Рисунок 3.2.16 – Завершение конвертации файла

Основные теги HTML это head, body, footer. Head это тег в котором происходит установка кодировки страницы, а также ссылки на стили применяемые для данной страницы (рисунок 3.2.17).

```

24 <head>
25 <title>Restofood</title>
26 <link rel="stylesheet" href="css/bootstrap.min.css">
27 <link rel="stylesheet" href="css/bootstrap-select.css">
28 <link href="css/style.css" rel="stylesheet" type="text/css" media="all" />
29 <link rel="stylesheet" href="css/flexslider.css" type="text/css" media="screen" />
30 <link rel="stylesheet" href="css/font-awesome.min.css" />
31 <!-- for-mobile-apps -->
32 <meta name="viewport" content="width=device-width, initial-scale=1">
33 <meta http-equiv="Content-Type" content="text/html; charset=UTF-8" />
34 <meta name="keywords" content="Restofood Responsive web template, Bootstrap Web Templates, Flat Web Templates, Android Compatible web template,
35 Smartphone Compatible web template, free webdesigns for Nokia, Samsung, LG, Sony Ericsson, Motorola web design" />
36 <script type="application/x-javascript"> addEventListener("load", function() { setTimeout(hideURLbar, 0); }, false); function hideURLbar(){
37     window.scrollTo(0,1); } </script>
38 <!-- //for-mobile-apps -->
39 <link href="//fonts.googleapis.com/css?family=Ubuntu+Condensed" rel="stylesheet" type="text/css">
40 <link href="//fonts.googleapis.com/css?family=Open+Sans:400,300,300italic,400italic,600,600italic,700,700italic,800,800italic" rel="stylesheet"
41     type="text/css">

```

Рисунок 3.2.17 – Тег head

```

79 <body>
80   <div class="header">
81     <div class="container">
82       <div class="logo">
83         <a href="index.html"><span>Resto</span>food</a>
84
85       </div>
86       <div class="header-right">
87         <a class="account" href="rp.php"><i class="fa fa-user" aria-hidden="true"></i> </a>
88         <a class="account" href="login.php">Личный кабинет</a>
89         <a class="account" href="exit.php">Выход</a>
90

```

Рисунок 3.2.18 – Тег header

На рисунке 3.2.18 показан тег body. Тег HTML <body> определяет основное содержимое HTML-документа или раздел HTML-документа, который будет непосредственно виден на вашей веб-странице. Этот тег также обычно называют элементом <body>. Деление на секции также удобно для прописывания стилей, которые будут применены к данным секциям.

На рисунке 3.2.19 показан код баннера, который видит юзер, который зашел на сайт. На данном баннере имеется ссылка добавления товара поставщиками, пользователь может с помощью кнопки «добавить товар» перейти на другую страницу сайта, где будет отображен код добавления товара (рисунок 3.2.20).

```

119   </div>
120   </div>
121 </div>
122 <div class="main-banner banner text-center">
123   <div class="container">
124     <h1>Restofood <span class="segment-heading"> приветствует </span> Вас!</h1>
125     <p>Удачного дня!</p>
126     <?php
127       if($_SESSION['typeaccount']=='postav')
128       {
129         echo "<a href='post-ad.php'>Добавить свой товар</a>";
130       }
131     ?>
132
133   </div>
134 </div>
135 </div>

```

Рисунок 3.2.19 – Код разметки баннера

```

173 <!-- Submit Ad -->
174 <div class="submit-ad main-grid-border">
175   <div class="container">
176     <h2 class="head">Добавить товар</h2>
177     <div class="post-ad-form">
178       <form action="" method="post">
179         <label>Выбрать категорию <span*></span></label>
180         <select id = "categoria" name = "categoria" onChange="this.form.submit();">
181           <option value='<? if($_POST['categoria']!='){ echo $_POST['categoria'];}>'><? if($_POST['categoria']!='){
if($_POST['categoria']==1){echo 'Мясо';}
182 if($_POST['categoria']==2){echo 'Овощи';}
183 if($_POST['categoria']==3){echo 'Фрукты';}
184 if($_POST['categoria']==4){echo 'Зелень';}
185 if($_POST['categoria']==5){echo 'Грибы';}
186 if($_POST['categoria']==6){echo 'Молочная продукция';}
187

```

Рисунок 3.2.20 – Код разметки добавления товара

```

1 <!DOCTYPE html>
2 <html>
3 <head>
4   <meta charset="utf-8">
5   <meta http-equiv="X-UA-Compatible" content="IE=edge">
6   <title>Restofood | Личный кабинет</title>
7   <!-- Tell the browser to be responsive to screen width -->
8   <meta name="viewport" content="width=device-width, initial-scale=1">
9   <!-- Font Awesome -->
10  <link rel="stylesheet" href="plugins/fontawesome-free/css/all.min.css">
11  <!-- Icons -->
12  <link rel="stylesheet" href="https://code.ionicframework.com/ionicons/2.0.1/css/ionicons.min.css">
13  <!-- Tempusdominus Bootstrap 4 -->
14  <link rel="stylesheet" href="plugins/tempusdominus-bootstrap-4/css/tempusdominus-bootstrap-4.min.css">
15  <!-- icheck -->
16  <link rel="stylesheet" href="plugins/icheck-bootstrap/icheck-bootstrap.min.css">
17  <!-- JQVMap -->
18  <link rel="stylesheet" href="plugins/jqvmap/jqvmap.min.css">
19  <!-- Theme style -->
20  <link rel="stylesheet" href="dist/css/adminlte.min.css">

```

Рисунок 3.2.21 – Разметка страницы личного кабинета

На рисунке 3.2.21 можно увидеть разметку страницы личного кабинета, также происходит разделение ролей поставщика и ресторатора.

Самой нижней частью страницы является footer. На нем обычно располагается разметка информации о компании, а также ссылки на соцсети или же другие способы связи. Результат разметки тега footer показан на рисунке 3.2.22.

```

322 <footer>
323 <div class="footer-top">
324 <div class="container">
325 <div class="foo-grids">
326 <div class="col-md-4 footer-grid">
327 <h4 class="footer-head">Жто мы?</h4>
328 <p>Благодаря Restofood вы можете совершать закупки напрямую без всяких переплат. Больше не нужно закупаться на несколько дней.
329 </p>
330 <p>Экономьте на транспорте. Экономьте на закупшке. Никаких перекупщиков.</p>
331 </div>
332 <div class="col-md-4 footer-grid">
333 <h4 class="footer-head">Информация</h4>
334 <ul>
335 <li><a href="index.php">Главная</a></li>
336 <li><a href="#">Личный кабинет</a></li>
337 <li><a href="categories.html">Категорик</a></li>
338 <li><a href="register.php">Регистрация</a></li>
339 </ul>
340 </div>
341 <div class="col-md-4 footer-grid">

```

Рисунок 3.2.22 – Нижняя часть страницы footer

Страница со всеми категориями платформы и используемые теги для разметки представлены на рисунке 3.2.23.

```

92 <h1>Добро пожаловать <span class="segment-heading"> в </span> категории!</h1>
93 <p>Удачного выбора!</p>
94 <a href="post-ad.html" hidden>Добавить товар</a>
95 </div>
96 </div>
97 <!-- Categories -->
98 <!--Vertical Tab-->
99 <div class="categories-section main-grid-border">
100 <div class="container">
101 <h2 class="head">Основные категории</h2>
102 <div class="category-list">
103 <div id="parentVerticalTab">
104 <ul class="resp-tabs-list hor_1">
105 <li>Мясо</li>
106 <li>Овощи</li>
107 <li>Фрукты</li>
108 <li>Зелень</li>
109 <li>Грибы</li>
110 <li>Молочная продукция</li>
111 <a href="all-classifieds.php">Показать все</a>

```

Рисунок 3.2.23 – Разметка страницы категорий

Как видно на рисунке 3.2.23 в разметке страницы категорий имеется 6 основных категорий, а также их подкатегории.

Более интересной страницей является карточка товара. На данной странице пользователь может увидеть созданный им товар, где отобразится название, цена, месторасположение, тип доставки, фотообложку и контакты. Разметка данной страницы представлена на рисунке 3.2.24.

```

124     </div>
125 </div>
126 <div class="banner text-center">
127     <div class="container">
128         <h1><span class="segment-heading">Карточка товара</span></h1>
129         <p>Удачной покупки!</p>
130
131     </div>
132 </div>
133 <!--single-page-->
134 <div class="single-page main-grid-border">
135     <div class="container">
136         <ol class="breadcrumb" style="margin-bottom: 5px;">
137             <li><a href="index.php">Главная страница</a></li>
138             <li><a href="all-classifieds.php">Все товары</a></li>
139             <?php
140                 $sql1 = "SELECT * FROM list_goods,Subsubcategory,Subcategory,Category,Goods WHERE list_goods.id_good='.$_GET['idgood'].'" AND
list_goods.subsub_id=Subsubcategory.subsub_id AND Subsubcategory.Subcategory_ID=Subcategory.Subcategory_ID AND
Subcategory.Category_ID=Category.Category_ID AND Category.Category_ID=Goods.Category_ID ";
141                 $result = mysqli_query($link, $sql1);
142                 $data = mysqli_fetch_array($result);

```

Рисунок 3.2.24 – Карточка товара

На рисунке 3.2.24 видно, что после заполнения была сформирована карточка товара в которой отображены все необходимые данные. Также реализованы функции редактирования товара, звонка и функции сравнения товаров с другими поставщиками.

Зайдя на страницу, где, в данном случае поставщик может увидеть «список желаний» ресторана, после чего может сформировать коммерческое предложение для нужного ресторана. Разметка всего этого процесса представлена на рисунке 3.2.25.

```

45 <body>
46 <div class="header">
47     <div class="container">
48         <div class="logo">
49             <a href="index.html"><span>Resto</span>food</a>
50         </div>
51         <div class="header-right">
52             <?php
53                 if($_SESSION['typeaccount']=='restoran'){
54                     echo '<a class="account" href="wishlist.php?idrest='.$_SESSION['typeaccount1'].'">Желания</a>';
55                 }
56             ?>
57
58             <a class="account" href="rp.php" <?php if(empty($_SESSION['id']))echo "hidden"; else{
59                 if ($_SESSION['typeaccount'] == 'postav') {
60                     echo "<a>Рестораны </a>";
61                 }
62                 if ($_SESSION['typeaccount'] == 'restoran') {
63                     echo "<a>Поставщики</a>";
64                 }

```

Рисунок 3.2.25 – Разметка списка желаний

Разметка для формы оформления заказа представлена на рисунке 3.2.26.

```

411     <button type="button" class="close" data-dismiss="modal" aria-label="закреть" >
412     <span aria-hidden="true"></span>
413     </button>
414     <h4 class="modal-title" id="myModalLabel">Оформление заказа</h4>
415 </div>
416 <div class="modal-body">
417     <form action="" method="post" onload="f();" >
418     <div class="flexslider">
419         <ul class="slides">
420             <?php
421
422                 $sql = "SELECT * FROM list_goods,Subsubcategory,Subcategory,Category,Goods WHERE
list_goods.id_good=".$GET['idgood']." AND list_goods.subsub_id=Subsubcategory.subsub_id AND Subsubcategory.Subcategory_ID=Subcategory.Subcategory_ID AND
Subcategory.Category_ID=Category.Category_ID AND Category.Category_ID=Goods.Category_ID AND list_goods.photo=Goods.ID_Good";
423                 $result = mysqli_query($link, $sql);
424                 $data = mysqli_fetch_array($result);
425
426                 echo '';?>
427             </ul>
428     </div>
429 </div>

```

Рисунок 3.2.26 – Разметка оформления заказа

Стили, которые применялись к тегам HTML документов представлены на рисунке 3.2.27.

```

7 body{
8     padding:0;
9     font-family: 'Open Sans', sans-serif;
10    background:#fff;
11 }
12 body a{
13     transition:0.5s all;
14     -webkit-transition:0.5s all;
15     -moz-transition:0.5s all;
16     -o-transition:0.5s all;
17     -ms-transition:0.5s all;
18     font-weight:400;
19     font-family: 'Open Sans', sans-serif;
20 }
21 input[type="button"],input[type="submit"]{
22     transition:0.5s all;
23     -webkit-transition:0.5s all;
24     -moz-transition:0.5s all;
25     -o-transition:0.5s all;
26     -ms-transition:0.5s all;

```

Рисунок 3.2.27 – Стили для тега body

На рисунке 3.2.27 показаны стили, примененные к тегу body, который взят в качестве селектора для объединения.

### 3.3 Описание клиентской части

HTML состоит из серии коротких кодов, набранных в текстовый файл автором сайта — это теги. Затем текст сохраняется в виде html-файла и просматривается через браузер, например Internet Explorer, Google Chrome, Mozilla Firefox. Этот браузер читает файл и переводит текст в видимую форму, надеясь, что страница будет выглядеть так, как задумал автор. Написание собственного HTML-кода влечет за собой правильное использование тегов для создания вашего видения. Для создания HTML-страниц можно использовать что угодно-от примитивного текстового редактора до мощного графического

редактора. Создать документы, написанные на языке HTML, которые станут основой вашего веб-сайта, можно несколькими способами. Теги - это то, что отделяет обычный текст от HTML-кода. Вы можете знать их как слова между <угловыми скобками> [26]. Они создают все интересные вещи, такие как изображения, таблицы и прочее, просто сообщая вашему браузеру, что нужно визуализировать на странице. Разные теги будут выполнять разные функции. Сами теги не появляются, когда вы просматриваете свою страницу через браузер, но их эффекты появляются.

Самым распространенным способом является текстовый набор HTML-кода в текстовом редакторе. Это довольно трудоемкая работа, которая подразумевает широкое познание принципов разметки документов в HTML, навыки работы с каскадными таблицами стилей CSS, а также как минимум с JavaScript, PHP и фреймворками.

Вторым способом является использование CMS. Система управления контентом, часто сокращенно называемая CMS - это программное обеспечение, которое помогает пользователям создавать, управлять и изменять контент на веб-сайте без необходимости специальных технических знаний. Проще говоря, система управления контентом - это инструмент, который помогает вам создавать веб-сайт без необходимости писать весь код с нуля (или даже знать, как кодировать вообще). Вместо того чтобы создавать свою собственную систему для создания веб-страниц, хранения изображений и других функций, система управления контентом обрабатывает все эти базовые инфраструктурные вещи для вас, чтобы вы могли сосредоточиться на более перспективных частях вашего веб-сайта. Помимо веб-сайтов, вы также можете найти системы управления контентом для других функций, таких как управление документами.

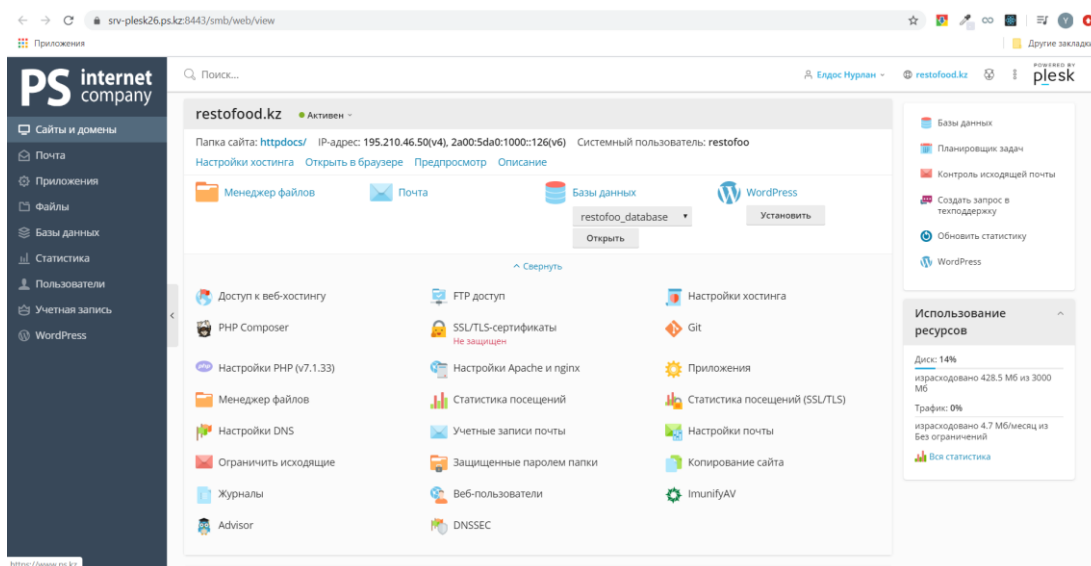


Рисунок 3.3.1 – система управления

Как описано выше, также для управления клиентской частью сайта используется инструмент для менеджмента файлов, редактирования кода, а также множество других функций, которые входят в панель управления Plesk (рисунок 3.3.1).

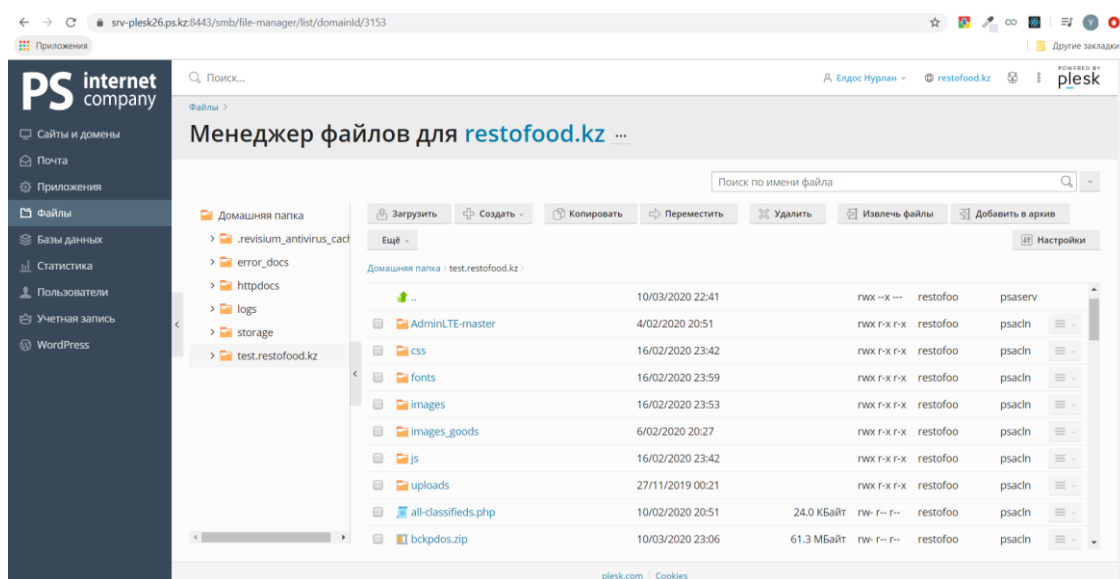


Рисунок 3.3.2 – менеджер файлов сайта

На рисунке 3.3.2 представлен менеджер управления файлами системы, с помощью него можно комфортно работать со всеми файлами и редактировать их внутренний код.

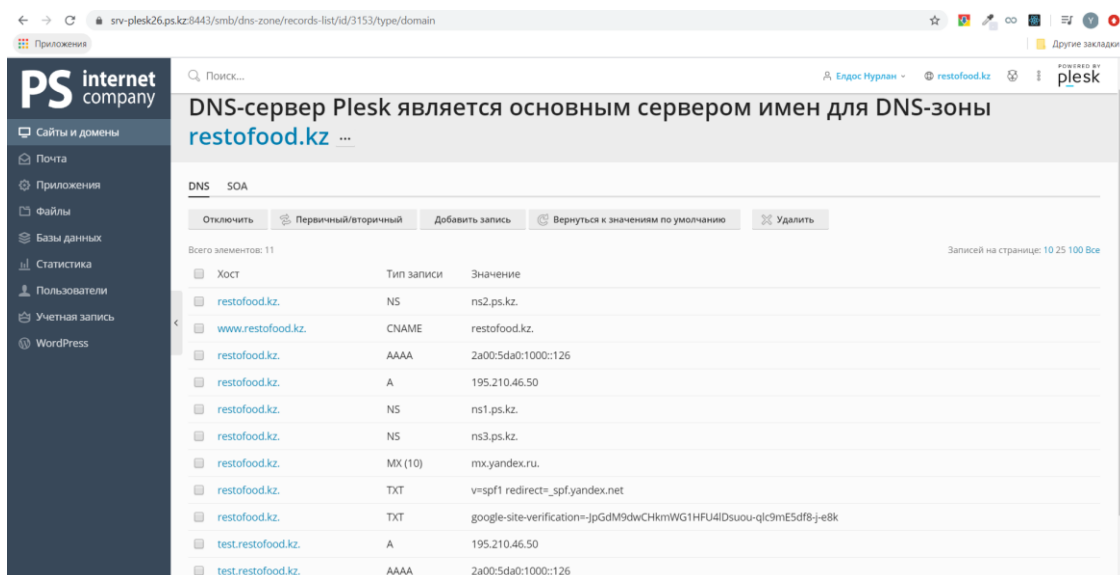


Рисунок 3.3.3 – настройки DNS

Страница с функцией управления DNS записями представлена на рисунке 3.3.3. Здесь отображаются NS, MX, TXT записи.



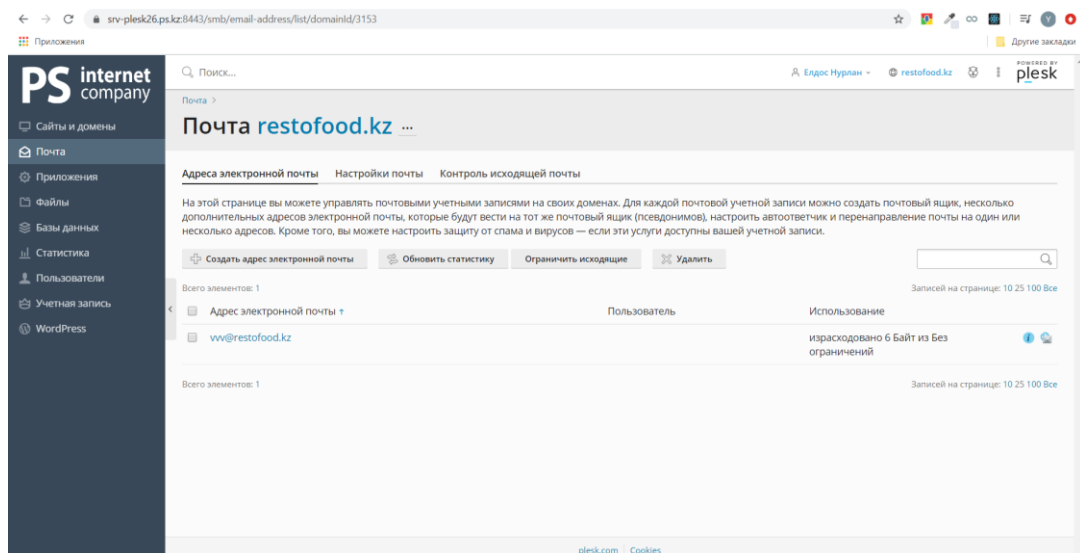


Рисунок 3.3.4 – учетные записи почты

На странице представленной на рисунке 3.3.4 вы можете управлять почтовыми учетными записями на своих доменах. Для каждой почтовой учетной записи можно создать почтовый ящик, несколько дополнительных адресов электронной почты, которые будут вести на тот же почтовый ящик (псевдонимов), настроить автоответчик и перенаправление почты на один или несколько адресов.

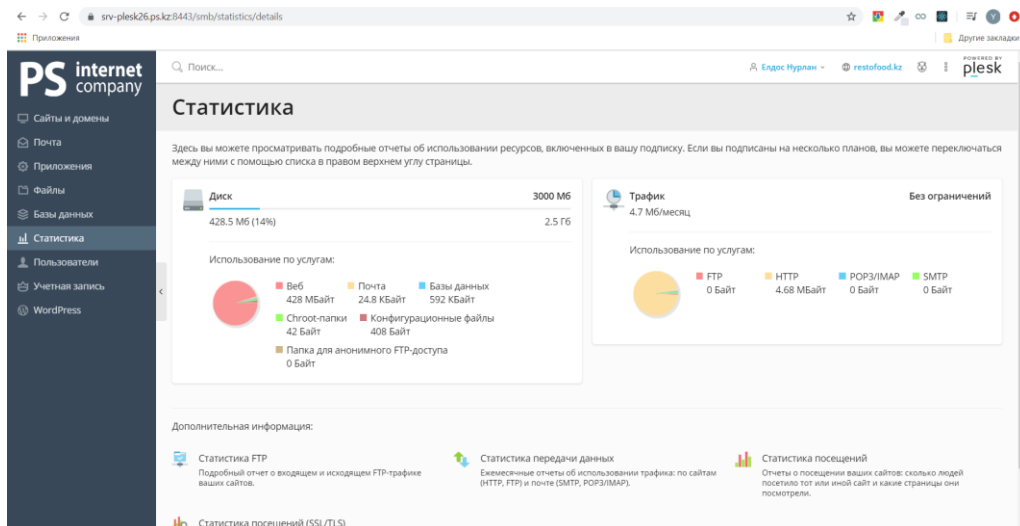


Рисунок 3.3.5 – статистика использования сайта

На рисунке 3.3.5 представлен просмотр подробных отчетов об использовании ресурсов, например, сколько место занято на виртуальном диске.

Также есть возможность создавать веб-сайты с помощью web-конструкторов. Создание веб-сайта раньше требовало обучения кодированию, но те времена остались в прошлом. Теперь любой желающий может легко

создать новый сайт с помощью конструктора веб-сайтов. Независимо от того, владеете ли вы небольшим бизнесом или хотите создать свой собственный блог, создатели сайтов делают это так, что вам не нужно нанимать веб-дизайнера, чтобы иметь профессионально разработанный сайт.

Но слышать, что конструкторы сайтов просты в использовании, не означает, что вы просто автоматически знаете, что такое конструктор сайтов или как он работает. Создатели веб-сайтов позволяют пользователям легко создавать свой собственный веб-сайт без особых усилий или навыков кодирования. Используя эту платформу, любой желающий может создать профессионально разработанный сайт без использования веб-дизайнера.

Для написания внешнего вида элементов применяется каскадные таблицы стилей CSS.

CSS расшифровывается как Каскадные Таблицы Стилей с акцентом на «стиль». В то время как HTML используется для структурирования веб-документа (определяя такие вещи, как заголовки и абзацы, а также позволяя вам вставлять изображения, видео и другие медиа), CSS определяет стиль вашего документа-макеты страниц, цвета и шрифты определяются с помощью CSS [4]. CSS привносит стиль на ваши веб-страницы, взаимодействуя с элементами HTML. Элементы - это отдельные HTML-компоненты веб-страницы - например абзац.

CSS имеет довольно простой синтаксис. Правила объявлений очень просты, и можно описать их в одном предложении. Во-первых пишется селектор, который выбирает конкретный элемент на странице после фигурных скобок пишутся свойства со значениями после двоеточия, а сами свойства отделяются друг от друга точкой с запятой. Самое сложное в CSS объявлении – селектор.

CSS селектор – (от слова select – выбирать) – это конструкция, с которой начинается каждый блок, и которая служит для выборки элемента или одинаковых элементов на странице для стилизации в будущем. Селекторы CSS используются для выбора контента, который вы хотите стилизовать. Селекторы являются частью набора правил CSS. Селекторы CSS выбирают HTML-элементы в соответствии с их идентификатором, классом, типом, атрибутом и т. д.

Важно учитывать, что когда пользователь заходит на сайт, его взгляд в первую очередь падает на дизайн. Дизайн играет очень важную роль, потому что от этого будет зависеть захочет ли юзер просматривать страницу дальше или закроет её. Для хорошего проектирования, разработки сайта и определения аудитории часто UX/UI.

UX относится к любому взаимодействию пользователя с продуктом или услугой. UX design учитывает каждый элемент, который формирует, как он заставляет пользователя чувствовать себя, и насколько легко для пользователя выполнить свои желаемые задачи. Это может быть что угодно - от того, как физический продукт ощущается в вашей руке, до того, насколько прост процесс оформления заказа при покупке чего-то онлайн. Цель UX design-

создать легкий, эффективный, актуальный и всесторонний приятный опыт для пользователя.

Дизайн пользовательского интерфейса (UI) - это процесс создания интерфейсов в программном обеспечении или компьютеризированных устройствах с акцентом на внешний вид или стиль. Дизайнеры стремятся создавать проекты, которые пользователи найдут простыми в использовании и приятными. Дизайн пользовательского интерфейса обычно относится к графическим пользовательским интерфейсам, но также включает в себя и другие, например управляемые голосом.

На нашей платформе интерфейс должен быть простым и понятным, так как данным сайтом будут пользоваться люди из областных центров города и закупщики ресторанов в возрасте от 18 до 70 лет.

Сайт будет состоять из главной страницы, личного кабинета, страницы категорий, карточки товаров, страницы добавления товара и модальных окон.



Рисунок 3.3.6 – Шапка сайта

На рисунке 3.3.6 продемонстрирована шапка сайта, в которой после авторизации пользователей отображаются списки (ресторанов, поставщиков), личный кабинет и выход с учетной записи. Также на рисунках 3.3.7 и 3.3.8 видно главную страницу где представлена шапка сайта, приветственный баннер, кнопка добавления товара, список категорий и футер сайта.



Рисунок 3.3.7 – Главная страница



Рисунок 3.3.8 – продолжение главной страницы

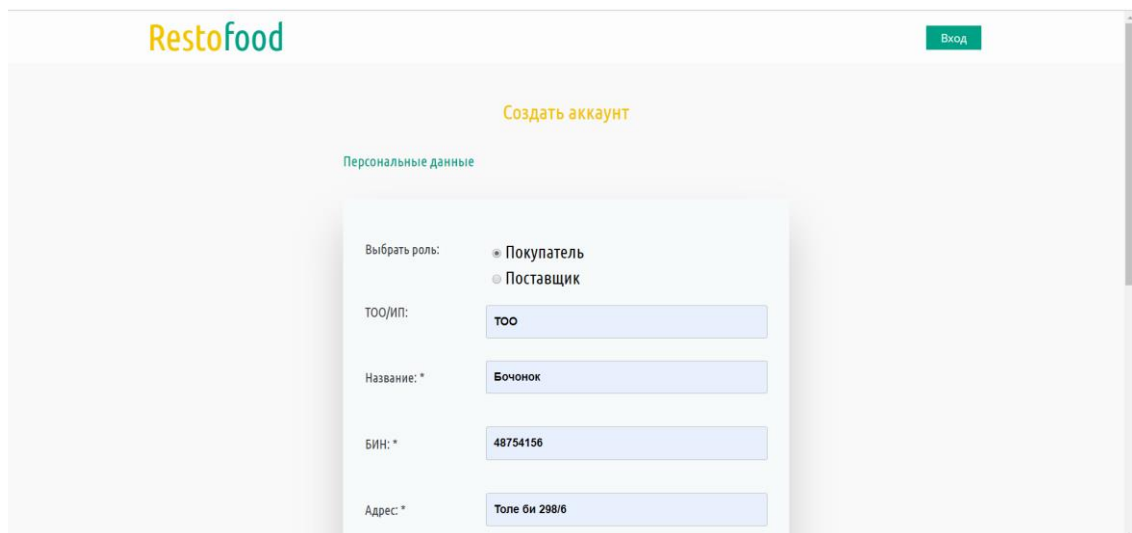


Рисунок 3.3.9 – Регистрация пользователей

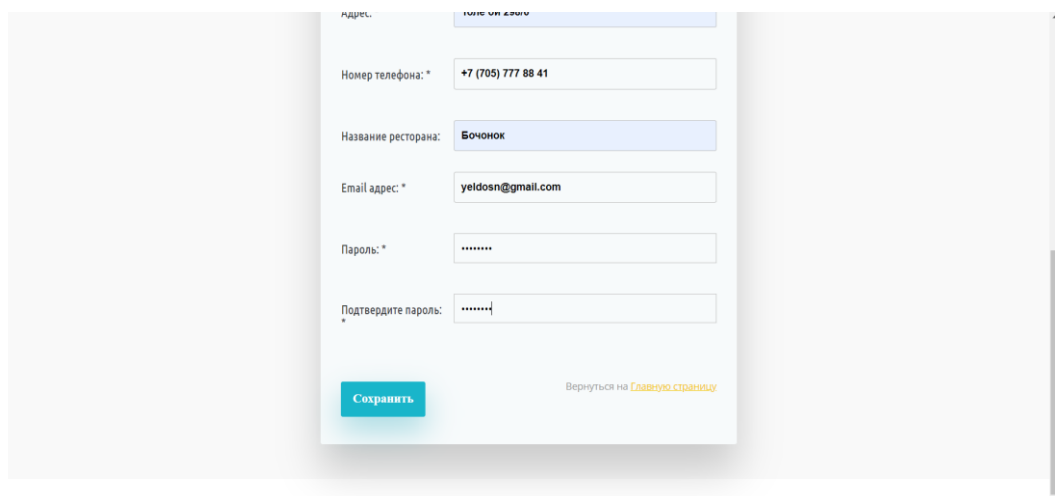
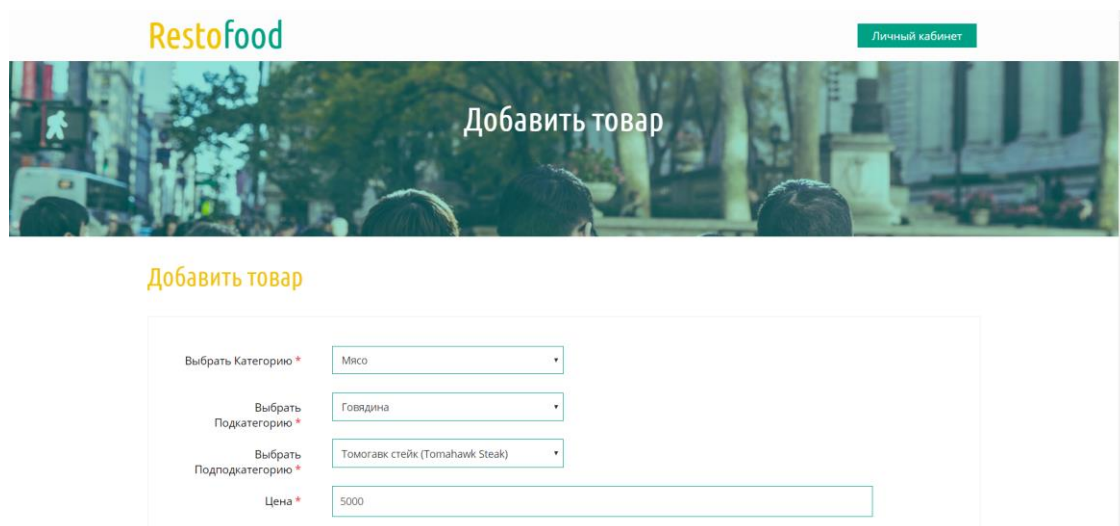


Рисунок 3.3.10 – продолжение страницы регистрации

Регистрация пользователей продемонстрирована на рисунках 3.3.9 и 3.3.10. То есть, перед тем как пользователь начнет пользоваться системой, ему необходимо зарегистрироваться, для использования функций платформы. На странице представлены шапка, баннер, и поля необходимые для регистрации пользователей.



The screenshot shows the Restofood website interface. At the top left is the logo 'Restofood' and at the top right is a button labeled 'Личный кабинет'. Below the header is a banner image with the text 'Добавить товар'. Underneath the banner is a form titled 'Добавить товар' with the following fields:

- Выбрать Категорию \* (Dropdown menu): Мясо
- Выбрать Подкатегорию \* (Dropdown menu): Говядина
- Выбрать Подподкатегорию \* (Dropdown menu): Томогак стейк (Tomahawk Steak)
- Цена \* (Text input): 5000

Рисунок 3.3.11 – Форма добавления товара



The screenshot shows the continuation of the 'Add Item' form. The fields are:

- Город \* (Text input): Алматы
- Наличие Сертификатов На Товар \* (Radio buttons):  Да,  Нет
- Описание Товара \* (Text area): Премиальная часть prime steak
- Доставка \* (Radio buttons):  Бесплатная,  Платная,  Самовывоз
- Более \* (Text input): 1 кг/л

Рисунок 3.3.12 – продолжение формы добавления товара

После нажатия кнопки «добавить товар» пользователь попадает на страницу добавления товара. На этой странице отображены шапка, баннер и поля для заполнения товара, такие как: категория, цена, месторасположение и другие (рисунок 3.3.11 и рисунок 3.3.12).

На рисунке 3.3.13 происходит разделение ролей поставщика и ресторатора, видно отображение личного кабинета с данными авторизованного пользователя. В личном кабинете при авторизации поставщика отображаются личные данные, его товары с функцией удаления, история заказов, переход на главную страницу. В личном кабинете ресторатора отображается все то же самое, но без редактирования товара.

Отображение товаров поставщика на платформе представлено на рисунке 3.3.14. На странице представлены все товары которые создал и добавил поставщик, также поставщик по желанию может их редактировать и удалять.

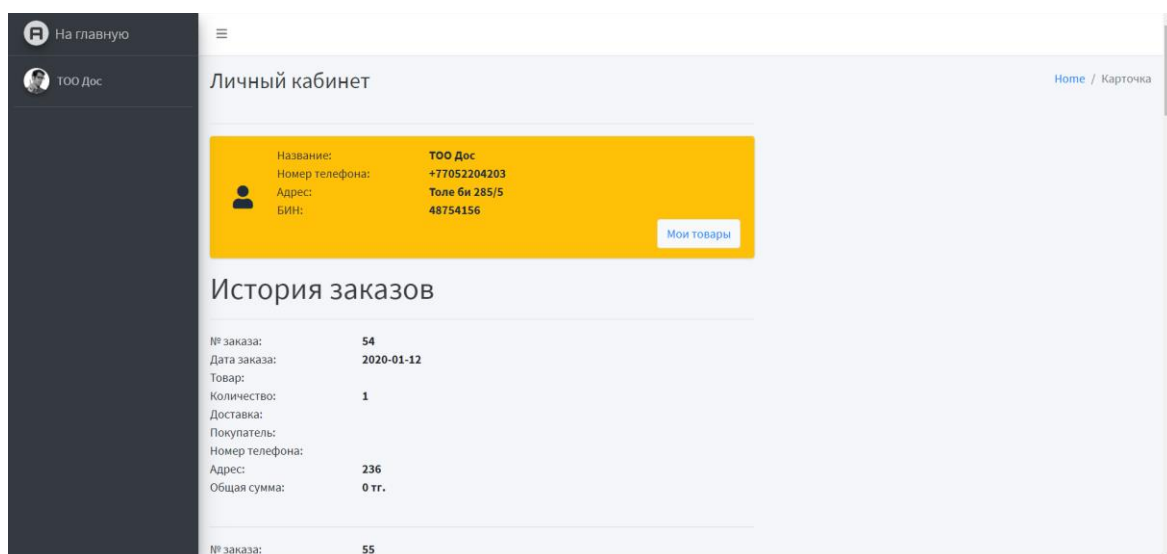


Рисунок 3.3.13 – Личный кабинет

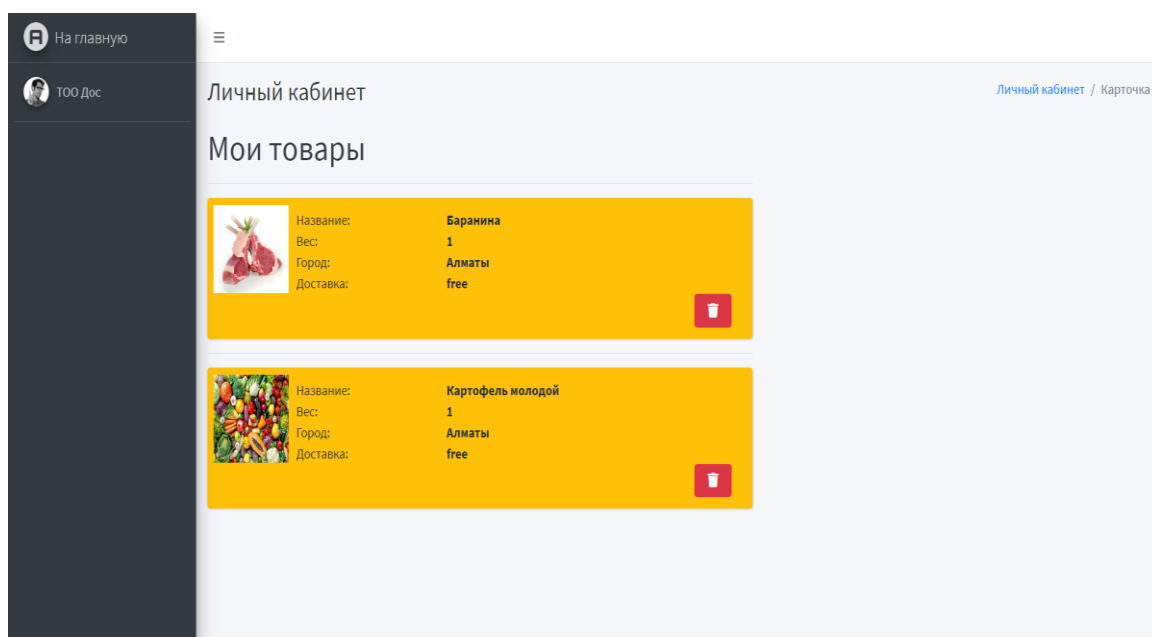


Рисунок 3.3.14 – Часть сайта footer

Как видно на рисунке 3.3.15 на странице категорий есть меню основных категорий с навигацией. Представлено 6 категорий (мясо, овощи, фрукты, зелень, грибы, молочная продукция), каждая категория делит свои подкатегории в виде невидимой сетки.

### Основные категории

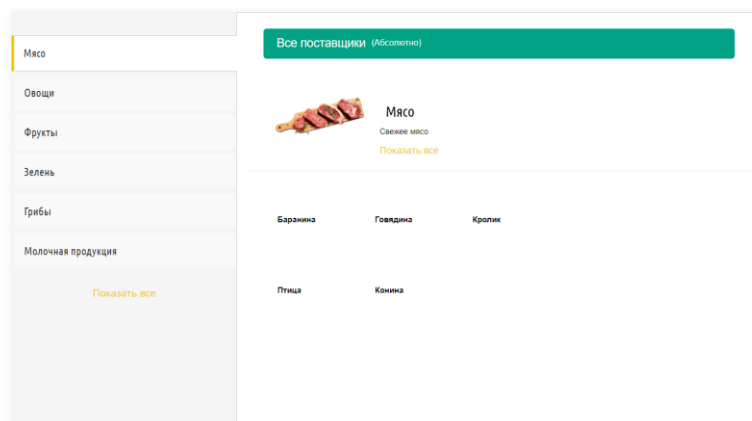


Рисунок 3.3.15 – Категории платформы

Страница карточки товара изображена на рисунке 3.3.16. На этой странице можно увидеть сформированную карточку. Здесь присутствует заголовок, дата добавления, фотообложка товара, описание, цена, месторасположение, поставщик, тип доставки, наличие сертификата, контактная информация. Также представлены функции отображения похожих товаров, редактирования товара и кнопка связи.

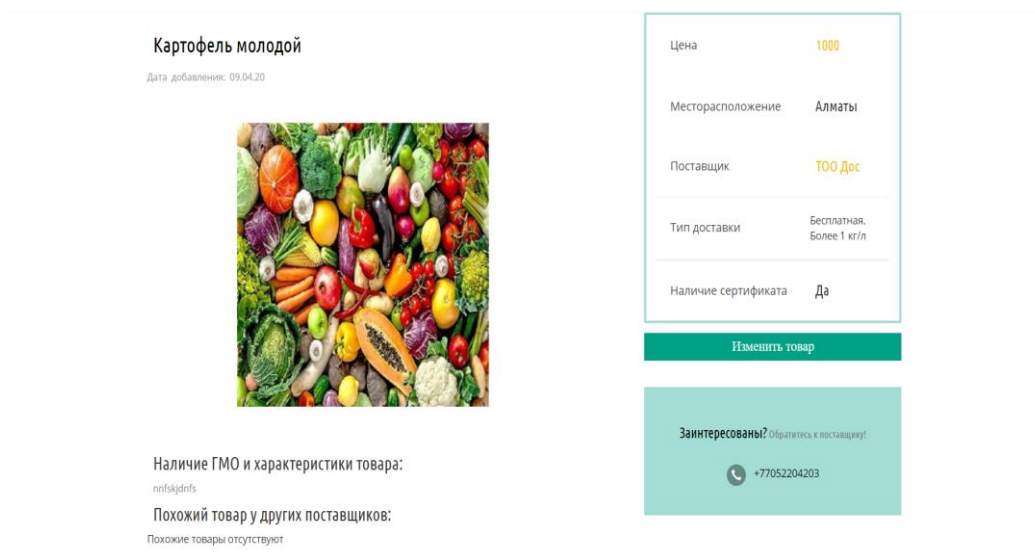


Рисунок 3.3.16 – Результат разметки карточки товара

Зайдя на страницу, где, в данном случае поставщик может увидеть «список желаний» ресторана, после чего может сформировать коммерческое предложение для нужного ресторана. То есть здесь присутствует шапка сайта, баннер и списки ресторанов с их данными для связи, также при нажатии на кнопку можно увидеть список «желаемого». Результат представлен на рисунке 3.3.17.

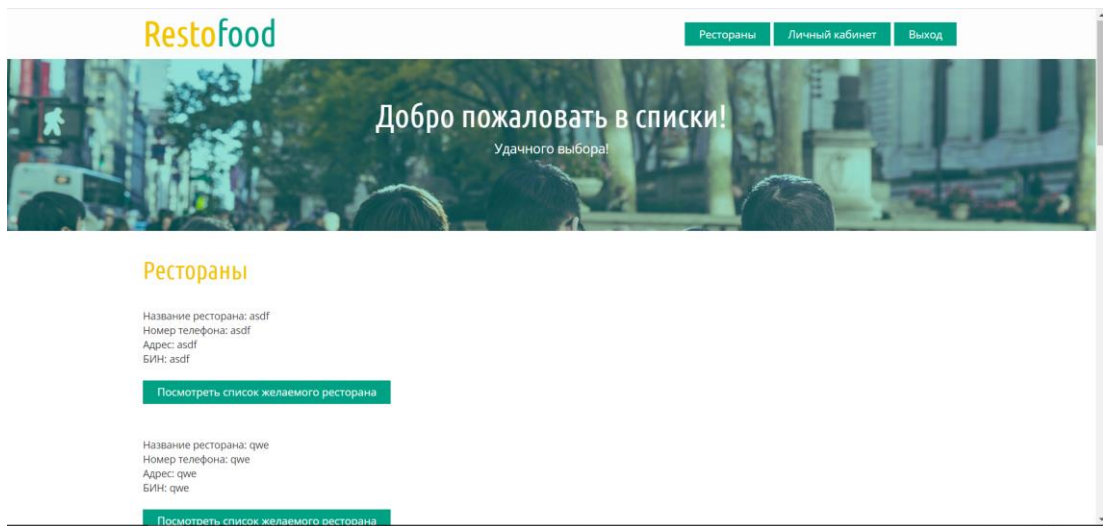


Рисунок 3.3.17 – Страница списка желаний

The image shows a modal window titled "Оформление заказа" with a close button (X) in the top right corner. In the center is a photograph of a bunch of yellow bananas. Below the photo, the following information is displayed: "Цена: 1 тг.", "Количество:" followed by an input field containing the number "1", "Доставка: free", and "Адрес:" followed by an empty input field. At the bottom right, it says "Итого: 1 тг." and there is a green button labeled "Заказать".

Рисунок 3.3.18 – Форма оформления заказа

Форма для удобного оформления заказа представлена на рисунке 3.3.18. В ней отображены фотообложка товара, цена, требуемое количество, адрес, итог и кнопка для заказа.



На рисунке 3.3.19 представлены товары которые добавил ресторан в список желаемого. В нем отображены все контактные данные ресторана, а также товары которые он добавил.

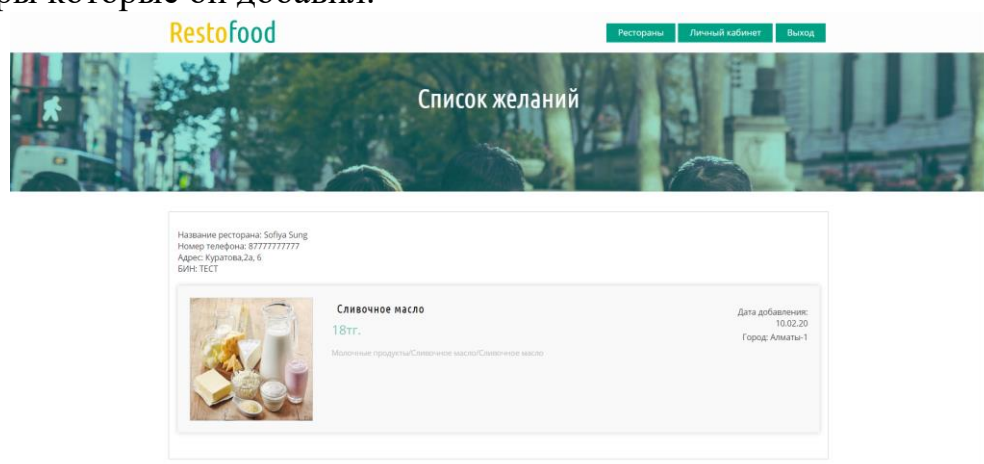


Рисунок 3.3.19 – список желаний

Отображение нижней части платформы (footer) представлено на рисунке 3.3.20. В ней представлена описательная часть о компании, навигация по сайту, контактная информация для связи, а также ссылки на социальные сети.

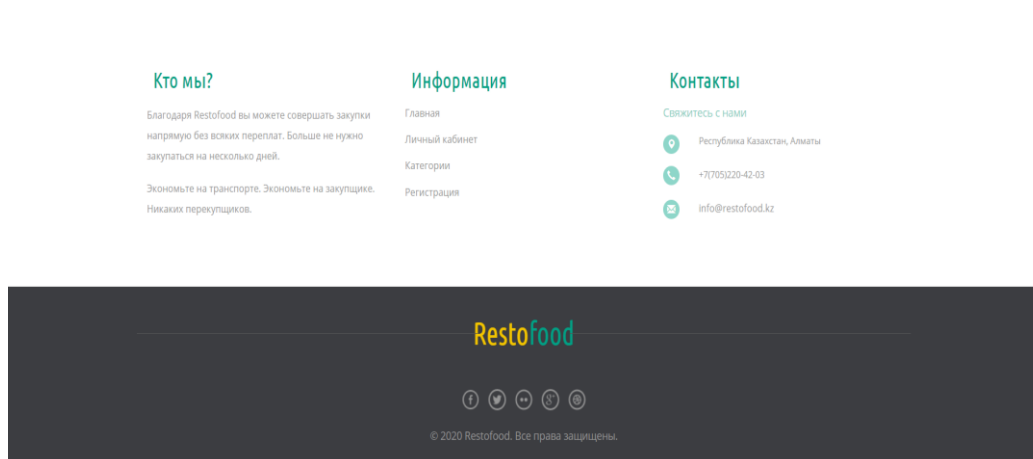


Рисунок 3.3.20 – footer сайта

### 3.4 Создание отдельной страницы для маркетинговой стратегии

На данный момент вопрос маркетинга является очень актуальным. В условиях мировой пандемии продолжит обостряться и борьба за удержание своих клиентов. Важной составляющей этой борьбы является проявление внимания к заказчикам, регулярное применение инструментов целевого маркетинга позволяет существенно уменьшить отток клиентов и сохранить прибыль в долгосрочной перспективе. Чтобы решить данную проблему,

дополнительно создана отдельная лендинг страничка, код которой представлен на рисунке 3.4.1.

```
469 </div>
470 <div class="hero-header">
471 <a href="#" class="link-block-2 w-inline-block">
472 <h1 class="heading-26">Resto Food</h1>
473 </a>
474 <h1 data-ix="hero-load-new" class="hero-heading"><span class="text-span-3">ЭКОНОМЬТЕ</span> НА ЗАКУПКЕ ПРОДУКТОВ ДЛЯ ВАШЕГО <strong>РЕСТОРАНА</strong>
</h1>
475 <p class="sub-header homepage" data-ix="hero-heading-loading">Наша компания осуществляет доставку продуктов в отели, рестораны и кафе. Команда
RestoFood имеет свою уникальную систему контроля качества, которая помогает поддерживать уровень сервиса наших постоянных партнеров. RestoFood это
стабильность поставок каждый день.</p>
476 <div class="div-block-2">
477 <a href="#contact" data-w-id="db33eaff-3313-9b05-3ad0-5a791aef8dc5" class="d-42-button w-inline-block">
478 <div data-w-id="db33eaff-3313-9b05-3ad0-5a791aef8dc6" style="-webkit-transform:translate3d(0, 0, 0) scale3d(0.1, 1, 1) rotateX(0) rotateY(0)
rotateZ(0) skew(0, 0);-moz-transform:translate3d(0, 0, 0) scale3d(0.1, 1, 1) rotateX(0) rotateY(0) rotateZ(0) skew(0, 0);-ms-transform:translate3d(0, 0,
0) scale3d(0.1, 1, 1) rotateX(0) rotateY(0) rotateZ(0) skew(0, 0);transform:translate3d(0, 0, 0) scale3d(0.1, 1, 1) rotateX(0) rotateY(0) rotateZ(0)
skew(0, 0)" class="d-42-inside-block"></div>
479 <div data-w-id="db33eaff-3313-9b05-3ad0-5a791aef8dc7" class="d-42-text-wrapper">
480 <div data-w-id="db33eaff-3313-9b05-3ad0-5a791aef8dc8" style="width:0PX" class="d-42-button-line"></div>
481 <div class="d-42-button-text">прайс</div>
482 </div>
483 </a>
```

Рисунок 3.4.1 – Разметка лендинга

Создана лендинг страничка для увеличения потока клиентов и популяризации платформы оптового закупа для ресторанов города Алматы. На ней представлена вся основная информация о компании, контактные данные, ассортимент продуктов, условия работы и т.д. Результат разметки показанный на рисунке 3.4.1 можно увидеть на рисунке 3.4.2.

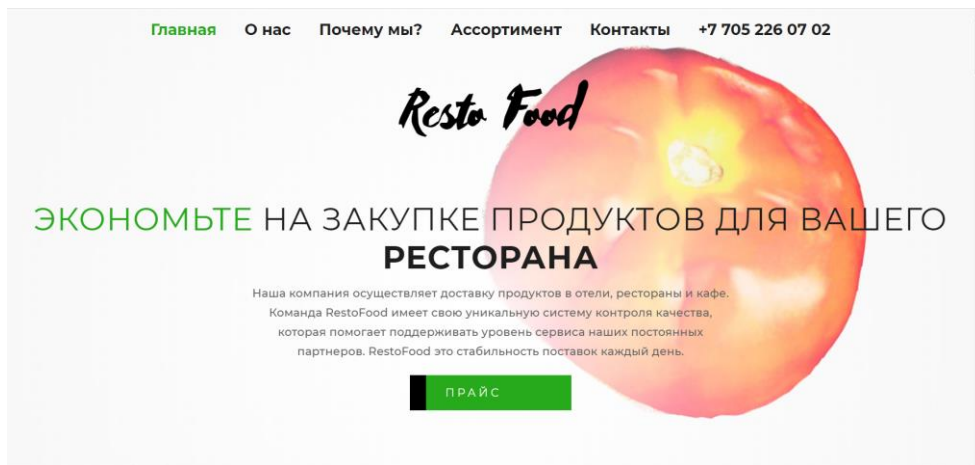


Рисунок 3.4.2 – Лендинг страничка

Также для максимального взаимодействия с пользователями и получения обратной связи от них, была разработана форма обратной связи, которая показана на рисунке 3.4.3.

Form for feedback (Форма обратной связи) with the following fields:

- ВАШЕ ИМЯ** (Your Name):
- ВАШ ТЕЛЕФОН** (Your Phone Number):
- ВАШЕ СООБЩЕНИЕ** (Your Message):
- (Send)

Рисунок 3.4.3 – Форма обратной связи

## 4 Экономический расчет

### 4.1 Цели и задачи, решаемые в экономической части

В рамках дипломного проекта описывается разработка автоматизированной системы заказа и продаж продуктов питания для ресторанов и поставщиков города Алматы. Web-платформа разрабатывается специально для ТОО «RestoFood(РестоФуд)» в целях упрощения процесса закупок для ресторанов и процесса продаж для поставщиков. В результате расчетов находится себестоимость разрабатываемой платформы.

Для расчета себестоимости необходимо учитывать:

- трудоемкость разработки программного продукта;
- материальные затраты;
- затраты на оплату труда;
- социальный налог;
- амортизация основных фондов;
- прочие затраты.

### 4.2 Расчет трудоемкости разработки ПП

Таблица 4.2.1 - Распределение работ по этапам, видам и оценка их трудоемкости

Этап разработки	Вид работы	Трудоемкость разработки, чел.× ч.
1	Анализ и описание задачи	40
2	Разработка алгоритма для ресторанов	60
3	Разработка алгоритма для поставщиков	75
4	Реализация клиентской части	150
5	Разработка администраторской части проекта	180
6	Настройка и тестирование программы	120
7	Оформление документации, инструкции пользователю, пояснительной записки.	70
ИТОГО		695

### 4.3 Расчет затрат на разработку ПП

Для определения затрат на разработку ПП производятся расчеты путем составления соответствующей сметы, которая включает в себя следующее:

- материальные затраты;
- затраты на оплату труда;

- социальный налог;
- амортизация основных фондов;
- прочие затраты.

Таблица 4.3.1 - Затраты на материальные ресурсы

Наименование материального ресурса	Единица измерения	Количество	Цена за единицу, тг	Сумма, тг
Ноутбук HP Pavilion Gaming 15-cx0113ur (5GV46EA W)	шт.	1	475000.00	475000.00
ИТОГО				475000.00

Общая сумма затрат на материальные ресурсы ( $Z_M$ ) определяется по формуле:

$$Z_M = \sum_{i=1}^n P_i * C_i, \quad (4.1)$$

где  $P_i$  - расход  $i$ -го вида материального ресурса, натуральные единицы;  
 $C_i$  - цена за единицу  $i$ -го вида материального ресурса, тг;  
 $i$  - вид материального ресурса;  
 $n$  - количество видов материальных ресурсов.

Если для разработки ПП используется электрооборудование, то необходимо рассчитать затраты на электроэнергию по форме, приведенной в таблице 4.3.2.

Общая сумма затрат на электроэнергию ( $Z_{\text{э}}$ ) рассчитывается по формуле:

$$Z_{\text{э}} = \sum_{i=1}^n M_i * K_i * T_i * C, \quad (4.2)$$

где,  $M_i$  - паспортная мощность  $i$ -го электрооборудования, кВт;  
 $K_i$  - коэффициент использования мощности  $i$ -го электрооборудования (принимается  $K_i=0.9$ );  
 $T_i$  - время работы  $i$ -го оборудования за весь период разработки ПП ч;  
 $C$  - цена электроэнергии, тг/кВт×ч;  
 $i$  - вид электрооборудования;  
 $n$  - количество электрооборудования.

Для расчета затрат на электроэнергию, необходимо найти данные исходя из продолжительности периода разработки ПП, количества кВт/ч, затраченных на проектирование ПП и тарифа за 1 кВт/ч. Тариф по городу

Алматы для юридических лиц в 2020 году составляет 19,17 тенге за 1 кВт/ч с учетом НДС (согласно данным представленным на официальном сайте ТОО «АлматыЭнергоСбыт»).

$$Z_э = 0,8 \cdot 0,9 \cdot 695 \cdot 19,17 \approx 9592,67 \text{ тг}$$

$$Z_э = 0,4 \cdot 0,6 \cdot 695 \cdot 19,17 \approx 3197,56$$

Таблица 4.3.2 - Затраты на электроэнергию

Наименование оборудования	Паспортная мощность, кВт	Коэффициент использования мощности	Время работы оборудования для разработки ПП, ч	Цена электроэнергии, тг/кВт·ч	Сумма, тг
Ноутбук HP Pavilion Gaming 15-sx0113ur (5GV46EAW)	0,8	0,9	695	19,17	9592,67
Освещение	0,4	0,6	695	18,17	3197,56
ИТОГО					12790,23

#### 4.4 Расчет затрат на оплату труда

Заработная плата Middle Full stack разработчика в 2020 составляет 250000 тенге (для города Алматы).

Рабочие часы сотрудника за месяц определяются по формуле:

$$Ч_м = N_м \cdot Ч_{рд}, \quad (4.3)$$

где,  $Ч_м$  — рабочие часы сотрудника за месяц;

$N_м$  — количество рабочих дней за месяц;

$Ч_{рд}$  — количество рабочих часов в день.

$$Ч_м = 21 \cdot 8 = 168 \text{ ч}$$

Ставка работника за час работы рассчитана по формуле:

$$ЧС_i = \frac{ЗП_i}{ФРВ_i}, \quad (4.4)$$

Инженер-разработчик:

$$\text{ЧС}_i = \frac{250000}{168} = 1488,09 \text{ тг}$$

где, ЗП<sub>і</sub> - заработная плата в месяц і-го работника, тг;  
ФРВ<sub>і</sub> - фонд рабочего времени в месяц і-го работника, час.

Для определения трудоемкости разработки ПП используются данные из таблицы 4.2.1.

Трудоемкость разработки Middle Full stack разработчика

$$T_2 = 40 + 60 + 75 + 150 + 180 + 120 + 70 = 695 \text{ чел.} \times \text{ч.}$$

Затраты на оплату труда в сумме (З<sub>тр</sub>) определяется по формуле:

$$З_{\text{тр}} = \sum_{i=1}^n \text{ЧС}_i * T_i, \quad (4.5)$$

где, ЧС<sub>і</sub> - часовая ставка і-го работника, тг;  
Т<sub>і</sub> - трудоемкость разработки ПП, чел.×ч;  
і - категория работника;  
п - количество работников, занятых разработкой ПП.

Инженер-разработчик:

$$З_{\text{тр}} = 1488,09 \cdot 695 = 1034222,55$$

Таблица 4.4.1 - Затраты на оплату труда

Квалификация	Трудоемкость разработки ПП, чел.×ч	Часовая ставка, тг/ч	Сумма, тг
Middle Full stack разработчик	695	1488,09	1034222,55
ИТОГО			1034222,55

Дополнительная заработная плата:

$$З_{\text{доп}} = З_{\text{тр}} \cdot 10\%, \quad (4.6)$$

$$З_{\text{доп}} = 1034222,55 \cdot 0,1 = 103422,255$$

Фонд заработной платы:

$$\Phi_{зп} = \Phi_{тр.о} + \Phi_{доп}, \quad (4.7)$$

$$\Phi_{зп} = 1034222.55 + 103422.255 = 1137644.8 \text{ тг}$$

Таблица 4.4.2 – Расчет социального налога

Налоги уплаченный юридическим лицом	10,46	ФОТ	1 137 645
СО (Социальные отчисления)	3,5	(ЗП - ОПВ)*3,5%	35 835,82
ВОСМСЮ (Отчисления на ВОСМСЮ)	2,0	ЗП*2%	22 752,90
СН (Социальный налог)	9,5	(ЗП - ОПВ - ВОСМСФ)*9,5%-СО	60 352,07
<b>Всего уплаченных налогов</b>			<b>118 941</b>

Расчет амортизационных основных фондов

Полная сумма амортизационных отчислений определяется по формуле:

$$З_{АМ} = \sum_{i=1}^n \frac{\Phi_i * H_{Ai} * T_{НИРi}}{100 * T_{ЭФi}}, \quad (4.9)$$

где,  $\Phi_i$  - стоимость  $i$ -го ОФ, тг;

$H_{Ai}$  - годовая норма амортизации  $i$ -го ОФ, %;

$T_{НИРi}$  - время работы  $i$ -го ОФ за весь период разработки ПП, ч;

$T_{ЭФi}$  - эффективный фонд времени работы  $i$ -го ОФ за год, ч/год;

$i$  - вид ОФ;

$n$  - количество ОФ.

Расчет годовой нормы амортизации ОФ:

$$H_{Ai} = \frac{100}{5} = 20$$

$$H_{Ai} = \frac{100}{T_{Ni}}, \quad (4.10)$$

где,  $T_{Ni}$  - возможный срок использования  $i$ -го ОФ, год;

Для определения продолжительности работы ПО для разработки ПП используются данные из таблицы 1.1.

Время работы ПО Sublime text и Plesk для разработки ПП составляет 585 часов (реализация клиентской части проекта, реализация администраторской части проекта, отладка и тестирование программы).



$$T_i = 60 + 75 + 150 + 180 + 120 = 585 \text{ ч.}$$

Оборудование:

$$Z_{AM} = \frac{475000 \cdot 20 \cdot 695}{100 \cdot 1920} = 34388.02 \text{ тг}$$

Таблица 4.4.3 - Амортизация основных фондов (ОФ)

Наименование оборудования и ПО	Стоимость оборудования и ПО, тг	Годовая норма амортизации, %	Эффективный фонд времени работы оборудования и ПО, ч/год	Время работы оборудования и ПО для разработки ПП, ч	Сумма, тг
Ноутбук HP Pavilion Gaming 15-сх0113ur (5GV46E A W)	475000	20	1920	695	34388.02
ОС Windows 10	Бесплатно	-	1920	695	-
ПО Sublime text	Бесплатно	-	1920	585	-
ПО Plesk	Бесплатно	-	1920	585	-
<b>ИТОГО</b>					<b>34388.02</b>

Таблица 4.4.4 - Смета затрат на разработку ПП

Статьи затрат	Сумма, тг	%
1. Материальные затраты, в том числе:		
- материалы	475 000.00	26,8
- электроэнергия	12790,23	0,72
2. Затраты на оплату труда	1137644.8	64,18
3. Отчисления на социальные нужды	118 941	6,35
4. Амортизация основных фондов	34388.02	1,94
5. Прочие затраты	0	

Продолжение таблицы 4.4.4

ИТОГО	1 772 449.89	100%
-------	--------------	------



Диаграмма 1 – Смета затрат на разработку проекта

#### 4.5 Определение возможной (договорной) цены ПП

Размер вероятной (договорной) стоимости ПП устанавливается с учетом производительности, свойства и сроков ее выполнения на уровне, отвечающем финансовым заинтересованностям клиента (потребителя) и исполнителя.

Договорная цена ( $Ц_d$ ) для прикладных ПП рассчитывается по формуле:

$$Ц_d = Z_{\text{НИР}} \left( 1 + \frac{P}{100} \right), \quad (4.11)$$

где,  $Z_{\text{НИР}}$  - затраты на разработку ПП (из таблицы 1.6), тг;

$P$  - средний уровень рентабельности ПП - 25 %

$$Ц_d = 1\,772\,449.89 * \left( 1 + \frac{25}{100} \right) = 1\,772\,449.89 + 443\,112.472 = 2\,215\,562.36 \text{ тг}$$

Дальше ориентируется стоимость реализации с учетом налога на добавленную цену (НДС), ставка НДС уточняется законодательно Налоговым Кодексом РК. На 2020 год ставка НДС установлена в размере 12%.

Цена реализации с учетом НДС рассчитывается по формуле:

$$Ц_p = Ц_d + Ц_d \cdot \text{НДС}, \quad (4.12)$$

$$Ц_p = 2215562.36 + 2215562.36 \cdot 0,12 = 2481429.89 \text{ тг}$$

В заключении нужно отметить что, цена реализации разработка автоматизированной системы заказа и продаж продуктов питания для ресторанов и поставщиков города Алматы для ТОО «RestoFood(РестоФуд)» с учетом всех возможных затрат составляет 2481429.89 тенге, себестоимость разработки равна 1 772 449.89, прибыль разработки равна 443112.472 тенге.

#### 4.6 Сравнительный анализ эксплуатационных затрат до и после внедрения

Статьи затрат после внедрения программного обеспечения:

- основная заработная плата разработчика и руководителя проекта с отчисляемыми налогами;
- материальные ресурсы;
- амортизация основных фондов;
- накладные расходы.

Статьи затрат до внедрения:

- основная заработная плата 1 заведующего складом, 1 бухгалтера и 2 закупщиков;
- материальные ресурсы;
- накладные расходы.

##### 4.6.1 Расчет затрат до внедрения ПО

Таблица заработной платы и отчисляемых налогов представлена в таблице 4.6.1.1

Таблица 4.6.1.1- Затраты на заработную плату

Сотрудник	Кол-во, чел	Заработная плата в месяц, тг	Заработная плата в год, тг	Социальное отчисление	Социальный налог	ВОСМС
Заведующий складом	1	200000	2 400 000	75 600	125 040	48 000
Бухгалтер	1	150 000	1 800 000	56 700	93 780	36 000
Закупщик	2	90 000 (180 000)	1 080 000 (2 160 000)	34 020 (68 040)	56 268 (112 536)	21 600 (43200)

К оплате	7 018 896
----------	-----------

Расчет затрат на необходимое для продукта ПО и оборудование представлен в таблице 4.6.1.2

Таблица 4.6.1.2 – Расчет затрат на оборудование и ПО, необходимое для проекта

Наименование материала	Марка	Ед. измерения	Количество	Цена за ед. в тенге	Сумма в тенге
Ноутбук	HP Pavilion Gaming 15-cx0113ur (5GV46EA W)	Штук	3	167000	501 000
Принтер	HP DeskJet 2130 (K7N77C)	Штук	1	41580	41580
Компьютерная мышь беспроводная	Logitech Laser Power 33458L	Штук	3	3 100	9300
Итого					551880

Затраты, отнесенные к амортизационным отчислениям представлены в таблице 4.6.1.3

Таблица 4.6.1.3-Амортизационные отчисления

Оборудование	Количество	Стоимость, тенге	Отчисления
Ноутбук HP Pavilion Gaming 15-cx0113ur (5GV46EA W)	3	501 000	13 176,3
Принтер HP DeskJet 2130 (K7N77C)	1	41 580	1 093,55
Итого	4	542 580	14 269,85

Общие накладные расходы вычисляются по формуле:

$$З_{н.р.} = 7\,018\,896 * 0,2 = 1\,403\,779,2 \text{ тенге.}$$

В итоге, общие эксплуатационные расходы после внедрения программного обеспечения будут равны,

$$З = 7\,018\,896 + 551\,880 + 14\,269,85 + 1\,403\,779,2 = 8\,988\,825 \text{ тенге.}$$

#### 4.6.2 Расчет затрат до внедрения ПО

Таблица заработной платы и отчисляемых налогов представлена в таблице 4.6.2.1

Таблица 4.6.2.1 - Затраты на заработную плату

Сотрудник	Кол-во, чел	Заработная плата в месяц одного, тг	Заработная плата в год, тг	Социальное отчисление	Социальный налог	ВОСМС
Разработчик	1	260 000	3 120 000	98 280	162 552	62 400
Руководитель	1	115 825	115 825	3648,5	6034,5	2 316,5
К оплате						3 571 056,5

Расчет затрат на необходимое для программного продукта и оборудование представлен в таблице 4.6.2.2

Таблица 4.6.2.2 – Расчет затрат на оборудование и ПО, необходимое для проекта

Наименование материала	Марка	Ед. измерения	Количество	Цена за ед. в тенге	Сумма в тенге
Ноутбук	HP Pavilion Gaming 15-cx0113ur (5GV46EAW)	Штук	1	167 000	167 000
Принтер	HP DeskJet 2130 (K7N77C)	Штук	1	41 580	51 580
Компьютерная мышь беспроводная	Logitech Laser Power 33458L	Штук	1	3 100	3 100
Итого					221680

Затраты, отнесенные к амортизационным отчислениям представлены в таблице 4.6.2.3

Таблица 4.6.2.3 - Амортизационные отчисления

Оборудование	Количество	Стоимость, тенге	Отчисления
Ноутбук HP Pavilion Gaming 15-cx0113ur (5GV46EA W)	1	167 000	4392,1
Принтер HP DeskJet 2130 (K7N77C)	1	41 580	1093,55
Итого	2	208 580	5485,65

Общие накладные расходы рассчитываются по формуле:

$$З_{н.р.} = 3\,571\,056,5 * 0,2 = 714\,211,3 \text{ тенге.}$$

Итого, общие эксплуатационные расходы до внедрения программного обеспечения будут равны,

$$З = 3\,571\,056,5 + 221\,680 + 5485,65 + 714\,211,3 = 4\,512\,433,45 \text{ тенге.}$$

Для более ясности предоставим все данные в сравнительную таблицу 4.6.2.4

Таблица 4.6.2.4 - Годовые эксплуатационные затраты

Статьи	До внедрения ПП	После внедрения ПП
Годовая заработная плата	7 018 896	4 512 433,45
Оборудование	551 880	221 680
Амортизационные отчисления	14 269,85	5485,65
Накладные расходы	1 403 779,2	714 211,3
Всего	8 988 825	4 512 433,45

Годовая экономия денежных средств будет равна согласно расчету:

$$8\,988\,825 - 4\,512\,433,45 = 4\,476\,392 \text{ тенге.}$$

Также определим суммарные выгоды от внедрения программного продукта.

Технико-экономические показатели позволяют рационально определить целесообразность разработки ПО и ее установки, а также взвесить реальную выгоду компании, как для разработчика системы, так и для ее пользователя.

Также можно определить срок окупаемости программного обеспечения в месяцах по формуле:

$$P_{\text{окуп}} = 12 * \frac{Z_{\text{разработка}}}{Э_{\text{годовая}}} \quad (4.13)$$

где  $P_{\text{окуп}}$  – период окупаемости в месяцах;

$Z_{\text{разр.}}$  – затраты на разработку программы;

$Э_{\text{годовая}}$  – годовая экономия.

$$P_{\text{окуп}} = 12 \cdot \frac{1\,772\,449.89}{4\,476\,392} = 12 \cdot 0,3959 = 4,7 \text{ мес.}$$

Предвидя годовую выгоду продукта и годовые эксплуатационные затраты с внедрением программного обеспечения можно определить коэффициент эффективности ( $K_{\text{эф.}}$ ) по формуле:

Расчетный коэффициент экономической эффективности капитальных вложений составляет:

$$E_p = \frac{Э_{\text{уг}}}{K}$$

$$E_p = \frac{4\,476\,392}{1\,772\,449,89} * 100\% = 253\% \text{ или } 2,53 \text{ тенге.}$$

Коэффициент эффективности показывает сколько экономии в тенге приходится на каждый тенге затрат. Можно заключить, что на каждый тенге затрат приходится 2,53 тенге экономии. Проводя аналогию, видно, что внедрение программного обеспечения будет благоприятствовать прибыльности компании, потому что установка позволит уменьшить количество кадров, что в свою очередь сократит выплаты заработной платы и налогов. В дополнение, сократится количество оборудования и амортизационных отчислений вместе с потребляемой электроэнергией. Программное обеспечение выработает более автоматизированный процесс, тем самым вызовет за собой сокращение материальных расходов и приумножит производительность труда.

## **5 Безопасность жизнедеятельности**

В данном дипломном проекте описывается разработка автоматизированной системы заказа и продаж продуктов питания для ресторанов и поставщиков города Алматы. Web-платформа разрабатывается специально для ТОО «RestoFood(РестоФуд)» в целях упрощения процесса закупок для ресторанов и процесса продаж для поставщиков. Чтобы обеспечить безопасность сотрудников был проведен анализ потенциально опасных и вредных факторов в офисе, воздействующих на персонал, а также выполнены расчеты обеспечивающие комфортные условия труда.

### **5.1 Анализ потенциально опасных и вредных факторов в офисе, воздействующих на персонал.**

На рабочем месте должны быть предусмотрены меры по защите от возможного воздействия опасных и вредных производственных факторов. Уровни этих факторов не должны превышать предельных значений, предусмотренных правовыми, техническими и санитарно-техническими нормами. Эти нормативные документы обязывают создавать условия труда либо на рабочем месте, при которых влияние опасных и вредных факторов на работников устраняется полностью, либо в допустимых пределах.

В процессе работы с компьютером необходимо соблюдать правильный режим работы и отдыха. В противном случае персонал замечает значительное напряжение зрительного аппарата с появлением жалоб на неудовлетворенность работой, головные боли, раздражительность, нарушение сна, усталость и боли в глазах, пояснице, шее и руках [18].

Работники сталкиваются с воздействием таких физически опасных и вредных производственных факторов, как повышенный уровень шума, повышенная температура окружающей среды, отсутствие или недостаточное освещение рабочей зоны, электрический ток, статическое электричество и другие.

Электробезопасность в лабораторных помещениях обеспечивается техническими методами и средствами защиты, а также организационно-техническими мероприятиями.

Рассмотрим основные причины травматизма человека электрическим током на рабочем месте:

1) Прикосновение к металлическим непроводящим частям (корпус, периферия компьютера), которые могут находиться под напряжением в результате повреждения изоляции;

2) Нерегулируемое использование электроприборов (запрещенное использование электроприборов, таких как электроплиты, чайники, нагреватели);

3) Отсутствие инструктажа сотрудников по правилам электробезопасности.



Во время работы на корпусе компьютера сохраняется статическое электричество. На расстоянии 5-10 см от экрана напряженность электростатического поля составляет 60-280 кв/м, то есть в 10 раз превышает норму 20 кв/м. Для снижения напряженности применяют увлажнители и нейтрализаторы, антистатические напольные покрытия [18].

Уровень шума на рабочих местах закупщиков и программистов видеоматериалов не должен превышать 50 дба, а в залах обработки информации на компьютерах – 65 дба. Для снижения уровня шума стены и потолки помещений, где установлены компьютеры, могут быть облицованы звукопоглощающими материалами. Уровень вибрации в помещениях вычислительных центров можно снизить, установив оборудование на специальные виброизоляторы. В таблице 5.1 можно ознакомиться с допустимыми нормами шума.

Таблица 5.1 – Допустимые нормы шума.

Виды производственной деятельности, рабочие места	Уровень звука, ДБА
1) Рабочие места в помещениях: дирекция, проектно — конструкторские бюро, программисты вычислительных машин, в лабораториях для теоретических работ	50
2) Рабочие места в помещениях цехового управленческого аппарата, в рабочих комнатах конторских помещений, лабораториях	60
3) Рабочие места в помещениях диспетчерской службы, машинописных бюро, на участках точной сборки, в помещениях мастеров, в залах обработки информации на вычислительных машинах	65
4) Рабочие места за пультами в кабинах наблюдения и дистанционного управления производственными циклами без речевой связи по телефону, в помещениях лабораторий с шумным оборудованием	75

*Продолжение таблицы 5.1*

5) Выполнение всех видов работ (за исключением п.1 — 4) на постоянных рабочих местах в производственных помещениях и на территории предприятия	80
--	----

Зоны с уровнем звука выше 85 ДБА должны быть обозначены знаками безопасности. Работающих в этих зонах администрация обязана снабжать средствами индивидуальной защиты. Шумовые характеристики машин или предельные значения шумовых характеристик должны быть указаны в паспортах на них, руководстве (инструкции) по эксплуатации.

К индивидуальным средствам защиты от шума относятся средства защиты органов слуха. На практике применяются беруши МАХ-1, МАХ-30, наушники противозумные ЗМ, ЕР-106, ЕР-107, ЕР-104.

К вредным производственным факторам могут быть отнесены физические факторы: электромагнитные поля и излучения, электростатические поля, постоянные магнитные поля, электрические и магнитные поля промышленной частоты (50 Гц), электромагнитные излучения радиочастотных диапазонов, электромагнитные излучения оптического диапазона (в том числе ультрафиолетовые, лазерные), ионизирующее излучение, тепловое (инфракрасное) излучение.

Основными источниками излучений при использовании энергии в информационных целях являются: персональные электронно-вычислительные машины (компьютеры), видео-дисплейные терминалы (мониторы), различные типы антенных устройств.

При работе закупщиков в сидячем положении необходимо соблюдать требования ГОСТ (ГОСТ 12.2.032-78.) к рабочему стулу пользователя:

- стул должен быть снабжен поворотным и подъемным механизмом;
- высота должна регулироваться;
- 380 мм глубины сиденья;
- 400 мм ширины сиденья;
- регулировать угол наклона спинки стула до 90-110 [23].

В первые этапы развития данная программа будет проходить тесты в помещениях выделяемыми рестораторами.

План помещения, где будет применяться разработанная программа, изображен на рисунке 5.1.1. В помещение в дневное время суток будут находиться 13 человек: один начальник закупок и 12 закупщиков. Закупщикам потребуются благоприятные обстоятельства труда, такие как рабочее пространство и положение внутренней среды комнаты, которое обеспечивает лучшую динамику работоспособности, не плохое здоровье и поддержание их самочувствия.

Здание, где сидят закупщики, содержит надлежащие габариты: длина -

14 м, ширина - 4 м, высота - 3 м, высота окошка - 2 м. Присутствует как натуральное, так например и искусственного происхождения освещение. Присутствует 2 окошка объемом 4x2 метров. Работники пространства размещаются между собой на расстоянии не менее 1,5 м. Каждое рабочее пространство гарантируется площадью 3 м<sup>2</sup>. В закупочном помещении располагается 13 личных лэптопов.

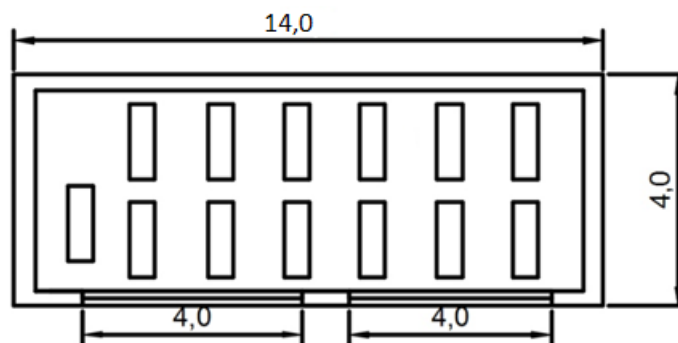


Рисунок 5.1.1 – План учебного помещения

Основа рационального планирования рабочего пространства - обеспечение сотрудников комфортабельными условиями для их продуктивной работы. Места рабочих для рабочего процесса оснащает соответствующий интерьер, отвечая наиболее благоприятным условиям работы.

## 5.2 Расчет естественного освещения

Делаем проверку площади боковых световых проветров здания, которая необходима для нормируемой освещенности на рабочей зоне.

Параметры пространства: длина  $L = 14$  м, ширина  $B = 4$  м, высота  $H = 3$  м. Высота рабочей поверхности над уровнем пола  $h_{рп} = 0,75$  м, окна начинаются с высоты  $h_{нок} = 0,8$  м, высота окна  $h_0 = 2$  м, ширина 4 м. Рабочее пространство находится в IV часовом поясе – в городе Алматы (пояс светового климата – IV северной широты и южнее).

Помещение, где будут, проходить работы, находится в  $l_{рт} = 1$  м, от наружной стенки здания, где задумываются оконные проветры. Наименьшая освещенность находится в точке, отстоящей на расстояние 4 м от оконного проема.

Общая площадь окон  $S_0, м^2$ , определяется по формулам 5.1 и 5.2

$$100 \frac{S_0}{S_n} = \frac{e_n \cdot K_3 \cdot \eta_0}{\tau_0 r_1}, \quad (5.1)$$

$$S_0 = \frac{S_n \cdot e_n \cdot K_3 \cdot \eta_0}{100 \cdot \tau_0 r_1}, \quad (5.2)$$

где  $S_n$ - площадь помещения;  $S_n = 56\text{м}^2$ ;

$e_n$  - нормированное значение КЕО, выбираемое по таблице 1.1. Для высокой точности зрительных работ принимаем  $e_n = 1,2$ .

$m_N$  - коэффициент светового климата, который находится по таблице 1.1 приложения. Учитывая заданный световой пояс, приняв ориентацию световых проемов на Север, определим по формуле 5.3, при  $m_N = 0,75$

$$e_x^{IV} = e_n \cdot m \cdot c, \quad (5.3)$$

где,  $m = 0,65$ ;

$c = 0,75$  (в наружных стенах зданий);

$e_n = 1,2$  для работ высокой точности III разряда зрительной работы;

$$e_x^{IV} = 0,65 \cdot 0,75 \cdot 1,2 = 0,585$$

Учитываем класс здания и находим коэффициент 5,3  $k_3 = 1,2$  (учебные помещения, лаборатории, конструкторские бюро).

$\tau_0$ - общий коэффициент светопропускания, равный

$$\tau_0 = \tau_1 \times \tau_2 \times \tau_3 \times \tau_4, \quad (5.4)$$

где,  $\tau_1 = 0,5$  (пустотелые стеклянные блоки);

$\tau_2 = 0,6$  (деревянные двойные раздельные переплеты);

$\tau_3 = 0,8$  (железобетонные фермы и арки);

$\tau_4 = 1$  (регулируемые жалюзи и шторы);

$\eta_0$  - световая характеристика окон.

$$\tau_0 = 0,5 \cdot 0,6 \cdot 0,8 \cdot 1 = 0,24$$

Определим по формуле 5.5

$$L = B - 1, \quad (5.5)$$

$$h_{\text{расч}} = h_{\text{ок}} + h_{\text{нок}} - h_{\text{рп}}$$

$$L = 4 - 1 = 3 \text{ м}$$

$$\frac{L}{1} = \frac{L}{B-1} = \frac{3}{3} = 1$$

$$h_{\text{расч}} = 0,75 + 2 - 0,75 = 2$$

$$\frac{B}{h_{\text{расч}}} = \frac{4}{2} = 2$$

Определим  $\eta_0 = 9$ .

$r_1$  – коэффициент, учитывающий повышение КЕО при боковом освещении благодаря свету, отраженному от поверхностей помещения и подстилающего слоя.

Средний коэффициент отражения в здании  $\rho_{\text{ср}} = 0,5$ , принимаем одностороннее боковое освещение. Определяем значение  $r_1$ . Принимаем  $r_1$  равным 1,05.

$k_{\text{зд}}$  – коэффициент, учитывающий затенение окон противостоящими зданиями.

Поскольку затеняющих зданий поблизости нет,  $k_{\text{зд}} = 1$ . Проведем расчет общей площади окон:

$$S_0 = \frac{56 \cdot 0,585 \cdot 9 \cdot 1 \cdot 1,2}{0,24 \cdot 1,05 \cdot 100} = 14,04 \text{ м}^2$$

Площадь световых проемов равна  $S_{\text{сп}} = 14,04 \text{ м}^2$ .

В ходе работы была рассчитана площадь боковых световых проемов закупочного помещения, необходимой для создания условий нормируемой освещенности на рабочих местах. Для здания размерами 14 х 4 х 3 м<sup>3</sup> значение для площади одного светового проема получилось равным 14,04 м<sup>2</sup>, где высота окна  $h_{\text{ок}} = 2$  м, а длина окна  $l_{\text{ок}} = 4$  м. В итоге, фактический размер светового проема удовлетворяет нормативным значениям.

### 5.3 Расчет искусственного освещения

Разряд зрительной работы – III. Нормируемая освещенность – 400 лк. В качестве светильника возьмем люминесцентные лампы PHILIPS STANDART. Длина светильника 80 см, ширина 20 см. Схема освещенности представлена на рисунке 5.3.1.

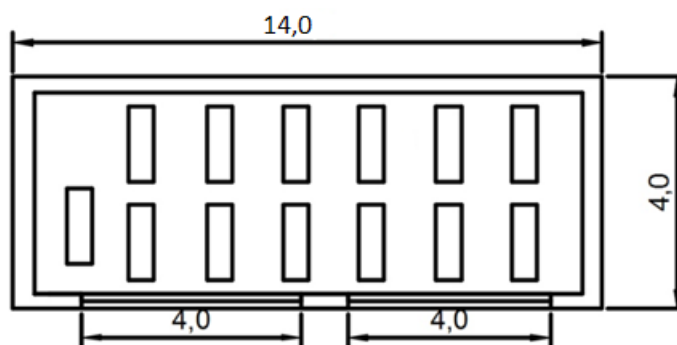


Рисунок 5.3.1 – Схема расчета освещенности.

Таблица 5.3 – Технические характеристики ламп PHILIPS STANDART

Тип	Мощность, Вт	Напряжение, В	Световой поток, лм	Длина, см	Ширина, см
PHILIPS STANDART	23	230	1500	80	20

Первым делом производим проверку соответствия минимальной заданной освещенности при имеющихся 6 светильников.

$$E = \frac{N \cdot n \cdot \Phi_{\text{л}} \cdot \eta}{k_z \cdot S_{\text{ос}} \cdot Z}, \quad (5.6)$$

где,  $S_{\text{ос}}$  – площадь помещения = 56;

$k_z$  – коэффициент запаса = 1,2;

$N$  – количество светильников = 2;

$Z$  – коэффициент неравномерности освещения,  $Z = 1,1$ ;

$n$  – количество ламп в светильнике = 4;

$\Phi_{\text{л}}$  – световой поток выбранной лампы,  $\Phi_{\text{л}} = 1500$  лм

$\eta$  – коэффициент использования,  $\eta = 60\%$

$$E = \frac{6 \cdot 2 \cdot 1500 \cdot 0,6}{1,2 \cdot 56 \cdot 1,1} \approx 146$$

При 2 светильниках освещенность примерно равна 146 лк, что не соответствует условиям труда. Значит, необходимо увеличить количество светильников для обеспечения приемлемой освещенности. Для этого нужно сделать расчеты для реконструкции:

Коэффициенты отражения от потолка стен и пола соответственно равны:

$$\rho_{\text{пот}} = 70 \%$$

$$\rho_{\text{ст}} = 50 \%$$

$$\rho_{\text{пол}} = 30 \%$$

Вычислим высоту подвеса светильника над рабочей поверхностью по формуле 5.6:

$$H = h - h_p - h_c, \quad (5.7)$$

где:  $h_c$  – расстояние от светильника до перекрытия,  $h_c = 0,1$  м;

$h_p$  – высота рабочей поверхности над полом,  $h_p = 0,6$  м;

$h$  – высота помещения,  $h = 3$  м.

$$H = 3 - 0,1 - 0,6 = 2,3 \text{ м}$$

Лучшее расстояние от окна до светильника определяется по формуле 5.8:

$$L = \lambda \cdot H, \quad (5.8)$$

где,  $\lambda = 0,6 \div 1,5$

$$L = 1,5 \cdot 2,3 = 3,45 \approx 3,5 \text{ м}$$

Расстояние между светильниками:

$$L_b = L_a - 1,5 = 3,5 - 1,5 = 2 \text{ м}$$

Промежуток от стены до ближайшего светильника, когда работа у стены не проводится, определяем по формуле 5.9

$$l_a = l_b = 3/3 = 1, \quad (5.9)$$

$$l_1 = 0,4 \cdot 3,2 = 1,28 \text{ м}$$

Устанавливаем индекс помещения по формуле 5.10

$$i = \frac{L \cdot B}{h_p \cdot (L + B)}, \quad (5.10)$$

$$i = \frac{4 \cdot 3}{2.175 \cdot (4 + 3)} = 0,79$$

Коэффициент использования в данном случае равен  $\eta = 60\%$ , коэффициент запаса равен  $k_3 = 1,2$

Установим количество люминесцентных ламп по формуле 5.10

$$N = \frac{E \cdot k_z \cdot S_{OC} \cdot Z}{n \cdot \Phi_{л} \cdot \eta} \quad (5.11)$$

где,  $S_{OC}$  – площадь помещения;

$k_z$  – коэффициент запаса;

$E$  – заданная минимальная освещенность,  $E = 400$  лк;

$Z$  – коэффициент неравномерности освещения,  $Z = 1,4$ ;

$n$  – количество ламп в светильнике;

$\Phi_{л}$  – световой поток выбранной лампы,  $\Phi_{л} = 1500$  лм

$\eta$  – коэффициент использования,  $\eta = 60\%$

$$N = \frac{400 \times 1,2 \times 56 \times 1,4}{4 \times 1500 \times 0,6} = 10,45 \approx 10$$

Всего для организации нормируемой освещенности 400 лк потребуется 10 люминесцентных ламп серии PHILIPS STANDART, мощность каждой лампы должна быть не менее 25 Вт, что соответствует действительности, следовательно освещения вполне хватит для соответствия санитарным нормам. Проведя расчеты, выяснили, что надо разрешить проблему с освещением и добавить еще 8 лампочек в помещение.

#### 5.4 Расчет систем кондиционирования рабочего помещения IT отдела.

Расчет систем кондиционирования требуется для приведения в соответствие температурного режима в помещениях нормам и создать благоприятные условия для закупщиков.

Выводимый воздух из пространства  $t_{\text{ВЫТ}}$ , °С,

$$t_{\text{ВЫТ}} = t_{\text{РЗ}} + \Delta t \cdot (h - z) \quad (5.12)$$

где  $t_{\text{РЗ}}$  – оптимальная температура для работы ( $t_{\text{РЗ}} \leq t_{\text{ДОП}}$ ), °С;

Для теплого периода года  $t_{\text{РЗ}} = 23$  °С;



$h$  - расстояние от пола до вытяжки кондиционера на потолке  $h = 3$  м;  
 $\Delta t$  - температурный градиент по высоте помещения, °С;  
 $z$  - высота рабочего помещения,  $z = 2,7$  м.

$$t_{\text{ВЫТ}} = 23 + 1,4 \times (3 - 2,7) = 23,4 \text{ °С}$$

Видно присутствие избытка явной теплоты. Поэтому для регулировки температурного режима и поддержке на благоприятном уровне, требуется подавать воздух ( $t_{\text{ПР}}$ ) на 5-7°С ниже температуры воздуха в пространстве.

$$t_{\text{ПР}} = 23 - 7 = 16 \text{ °С}$$

Произвели калькуляцию количества приточного воздуха  $L_{\text{ПР}}, \text{м}^3/\text{ч}$ :

$$L_{\text{ПР}} = \frac{Q_{\text{ИЗБ}}}{c \cdot \rho_{\text{ПР}} \cdot (t_{\text{ВЫТ}} - t_{\text{ПР}})} \quad (5.13)$$

где  $\rho_{\text{ПР}} - 1,2 \text{ кг}/\text{м}^3$  - плотность поступающего в помещение воздуха;  
 $c - 1 \text{ кДж}/(\text{кг} \times \text{°С})$  – удельная теплоемкость воздуха при постоянном давлении;

$t_{\text{ВЫТ}}$  - температура воздушной массы, удаляемой из рабочего пространства, °С;

$Q_{\text{ИЗБ}}$  – избыточное выделение явной теплоты, кДж/ч;

$t_{\text{ПР}}$  - температура приточного воздуха в рабочее помещение, °С.

Определили величину избыточного выделения явной теплоты  $Q_{\text{ИЗБ}}$ :

$$Q_{\text{ИЗБ}} = \sum Q - \sum Q_{\text{УХ}} \quad (5.14)$$

где  $\sum Q$  - суммарное количество поступающей явной теплоты из вне и выделяемой явной теплоты внутри помещения;

$\sum Q_{\text{УХ}}$  - суммарное количество выводящейся из пространства теплоты.

Тепловыделения от искусственного освещения  $Q_2$ :

$$Q_2 = 1000 \times N \quad (5.15)$$

$N$  – Мощность освещения, Вт.

$$Q_2 = 1000 \times 0,25 \times 20 = 5000 \text{ Вт}$$

Тепловыделение от людей  $Q_3$ :

$$Q_3 = n \times q \quad (5.16)$$

где  $n$  - число сотрудников;

$q$  - количество тепла, от каждого человека, Вт.

Таблица 5.4.1 - Количество тепла, от одного человека в зависимости от категории работ и температуры окружающей среды.

Категория выполняемых работ	Выделяемое тепло			
	Полное		Явное	
	10 °С	35 °С	10 °С	35 °С
Легкая	160 Вт	140 Вт	155 Вт	135 Вт

$$Q_3 = 21 \times 140 = 2\,940 \text{ Вт}$$

Количество тепла от солнечной радиации  $Q_{\text{ОСТ.РАД}}$ :

$$Q_{\text{ОСТ.РАД}} = F_{\text{ОСТ}} \times q_{\text{ОСТ}} \times A_{\text{ОСТ}} \quad (5.17)$$

Количество тепла от солнечной радиации в зависимости от наружного покрытия:

$$Q_{\text{П.РАД}} = F_n \times q_n \times k_n \quad (5.18)$$

где  $F_{\text{ОСТ}}$  и  $F_n$  - площадь через которое поступает солнечная радиация,  $\text{м}^2$ ;  
 $q_{\text{ОСТ}}$  и  $q_n$  - теплопоступления через  $1 \text{ м}^2$  поверхности. коэффициент теплопередачи равен  $1 \text{ Вт}/(\text{м}^2 \times ^\circ\text{С})$ ;

$A_{\text{ОСТ}}$  - коэффициент остекления рабочего помещения;

$k_n$  - коэффициент теплопередачи покрытия,  $\text{Вт}/(\text{м}^2 \times ^\circ\text{С})$ .

$q_{\text{ОСТ}}$  - принимается в пределах 70 – 210;

$A_{\text{ОСТ}}$  – принимается в пределах 0,25 - 1,25.

$$F_{\text{ОСТ}} = 50 \text{ м}^2$$

$$q_{\text{ОСТ}} = 150 \text{ Вт}/(\text{м}^2 \times ^\circ\text{С});$$

$A_{\text{ОСТ}} = 0,4$  – окна в пространстве не затемненные;

$$Q_{\text{ОСТ.РАД}} = 50 \times 150 \times 0,4 = 3\,000 \text{ Вт}$$

$Q_{\text{П.РАД}} = 18 \text{ Вт}$  - среднее теплопоступление покрытия;

$$Q_{\text{П.РАД}} = F_n \times q_n \times k_n$$

$$Q_{\text{П.РАД}} = 50 \times 1 \times 18 = 900 \text{ Вт}$$

Количество тепла которое выходит из рабочего пространства  $Q_{\text{УХ}}$ , кВт:

$$Q_{\text{УХ}} = \frac{\lambda \cdot S \cdot (t_{\text{ВЫТ}} - t_{\text{ПР}})}{\delta} \quad (5.19)$$

где  $\lambda$  - теплопроводность стен, Вт/(м × °С);

$S$  - площадь, м<sup>2</sup>;

$\delta$  - толщина стен, м;

$\delta = 0,3\text{м}$  толщина стен, которые выполнены из бетона К750 с теплопроводностью 1,2 Вт/(м × °С).

$$Q_{\text{УХ}} = \frac{1,2 \cdot 330 \cdot (23,4 - 16)}{0,3} = 9\,768 \text{ Вт}$$

Сумма входящей теплоты в помещение:

$$\sum Q = Q_2 + Q_3 + Q_{\text{ОСТ.РАД}} + Q_{\text{П.РАД}} \quad (5.20)$$

$$\sum Q = 5000 + 2940 + 3000 + 90 = 11\,030 \text{ Вт}$$

Избыточное выделение явной теплоты:

$$Q_{\text{ИЗБ}} = 11\,030 - 9\,768 = 1\,262$$

Количество приточного воздуха необходимого для поддержания комфортных условий:

$$L_{\text{ПР}} = \frac{1262}{1 \cdot 1,2 \cdot (23,4 - 16)} = 142,12 \frac{\text{м}^3}{\text{ч}}$$

Так как имеется избыточное выделение теплоты, рассчитали, что для приведения условий труда в комфортные и соответствию нормам условия нужно установить кондиционер.

Выбрали кондиционер LG СТ24 - инверторный кассетный кондиционер с сплит – системой и возможностью как обогрева, так и охлаждения.

Таблица 5.4.2 Характеристики сплит-системы Gree-42

1	Мощность в режиме охлаждения	13 000 Вт
2	Мощность в режиме обогрева	14 600 Вт
3	Потребляемая мощность при охлаждении	3 200 Вт

4	Потребляемая мощность при обогреве	3 000 Вт
5	Уровень шума (мин/макс)	33/37 Дб

*Продолжение таблицы 5.4.2*

6	Обслуживаемая площадь	130 кв.м
7	Внутренний блок сплит-системы	900×900×375 мм
8	Наружный блок сплит-системы	1150×550×1210 мм

## Заключение

В ходе выполнения дипломного проекта была разработана программа которая позволит управлять рисками для поддержки бизнес процессов компании. В рамках проекта было решено разработать автоматизированную систему заказа и продаж продуктов питания для ресторанов и поставщиков города Алматы, так как риски были в сегменте общественного питания.

При реализации дипломного проекта были выполнены следующие задачи:

- была разработана автоматизированная система заказа и продаж продуктов питания для ресторанов и поставщиков города Алматы;
- определены риски угрожающие компаниям;
- устранены риски для бизнес процессов;
- открыт дополнительный канал продаж для поставщиков;
- организовано чистое ценообразование;
- создана маркетинговая страничка, а также повышена работоспособность и эффективность предприятий.

Было произведено экономическое обоснование целесообразности разрабатываемого программного продукта, где было сделано следующее заключение: цена реализации окупает все затраты, потрачены на разработку. Прибыль от реализации проекта равна 443112.472 тенге;

Также были исследованы условия труда в помещении, где возможно будет применяться программа, проведен анализ потенциально опасных и вредных факторов в офисе, воздействующих на персонал и были предложены мероприятия по улучшению качества освещения в помещении, а также был произведен расчет систем кондиционирования.

## Список литературы

- 1 Научная интернет библиотека Киберленинка URL:  
<https://cyberleninka.ru/article/n/sostoyanie-i-tendentsii-razvitiya-rynka-obshchestvennogo-pitaniya-regiona>
- 2 Научная интернет библиотека Киберленинка URL:  
<https://cyberleninka.ru/article/n/analiz-rynka-i-klassifikatsiya-predpriyatii-obshchestvennogo-pitaniya>
- 3 Научная интернет библиотека Киберленинка URL:  
<https://cyberleninka.ru/article/n/osnovnye-tendentsii-razvitiya-restorannogo-biznesa-v-kazahstane-v-dokrizisnyu-i-krizisnyu-period>
- 4 Andy, Harris HTML, XHTML and CSS All–In–One For Dummies® / Andy Harris. - Москва: Наука, 2014. - 173 с.
- 5 Ben, Henick HTML & CSS – The Good Parts / Ben Henick. - Москва: СИНТЕГ, 2013. - 350 с.
- 6 Ed, Tittel HTML, XHTML & CSS For Dummies® / Ed Tittel. - Москва: Гостехиздат, 2012. - 416 с.
- 7 Советов Б. Я. Архитектура информационных систем / Б. Я. Советов, А. И. Водяхо, В. А. Дубенецкий, В. В. Цехановский. – М.: Академия, 2012. 228с.
- 8 Зандстра М. PHP. Объекты, шаблоны и методики программирования./М. Зандстра– М: Вильямс, 2011. 560с.
- 9 Чудновский А.Д. Жукова М.А. Информационные технологии управления в туризме: учебное пособие.-М.:Кнорус, 2010 -101 с.
- 10 Хольцшлаг, Молли Э. Использование HTML 4. Специальное издание (+ CD - ROM) / Хольцшлаг Молли Э.. - М.: Вильямс, 2012. - 174 с.
- 11 Харрис Энди PHP/MySQL для начинающих; КУДИЦ-Образ - М., 2016. - 384 с.
- 12 Дронов, В. HTML 5, CSS 3 и Web 2.0. Разработка современных Web-сайтов / В. Дронов. - М.: БХВ-Петербург, 2014. - 138 с.
- 13 Дронов, Владимир HTML 5, CSS 3 и Web 2.0. Разработка современных Web-сайтов / Владимир Дронов. - М.: БХВ-Петербург, 2013. - 416 с.
- 14 Бенкен, Е. С. PHP, MySQL, XML. Программирование для Интернета (+ CD-ROM) / Е.С. Бенкен. - М.: БХВ-Петербург, 2011. - 304 с.
- 15 Зандстра Мэтт PHP. Объекты, шаблоны и методики программирования; Вильямс - М., 2016. - 560 с.
- 16 Колисниченко Денис PHP и MySQL. Разработка Web-приложений; БХВ-Петербург - М., 2017. - 560 с.
- 17 Квинт, Игорь Создаем сайты с помощью HTML, XHTML и CSS / Игорь Квинт. - М.: Питер, 2014. - 448 с.
- 18 Основы безопасности жизнедеятельности: Учебное пособие / сост.: М.К.Дюсебаев, Ж.С.Абдимуратов. -Алматы: АУЭС, 2013.-79 с

19 «HTML и CSS. Разработка и дизайн веб-сайтов» / Джон Дакетт; [пер. с англ. М. А. Райтмана]. – Москва: Эксмо, 2019. -480 с.: ил. - (Мировой компьютерный бестселлер)

20 Никсон Робин Создаем динамические веб-сайты с помощью PHP, MySQL, JavaScript и CSS; Питер - М., 2017. - 204 с.

21 Справочник экономики URL: <http://ru.solverbook.com/spravochnik/formuly-po-ekonomike/formula-obshhej-rentabelnost/>

22 «JavaScript и jQuery. Интерактивная веб-разработка» / Джон Дакетт; [пер. с англ. М. А. Райтмана]. — Москва: Эксмо, 2017. — 640 с.: ил. — (Мировой компьютерный бестселлер).

23 Документ ГОСТ 12.2.032-78 Система стандартов безопасности труда (ССБТ). Рабочее место при выполнении работ сидя.

24 Бардзелл, Джеффри Macromedia Dreamweaver MX 2004 с ASP, ColdFusion и PHP. Из первых рук (+ CD-ROM) / Джеффри Бардзелл. - М.: Эком, 2017. - 560 с.

25 Бенкен, Е. PHP, MySQL, XML. Программирование для Интернета / Е. Бенкен. - М.: БХВ-Петербург, 2016. - 352 с. Wikipedia: Парковочные часы / URL: [https://en.wikipedia.org/wiki/Parking\\_meter](https://en.wikipedia.org/wiki/Parking_meter)

26 Прохоренок, Николай HTML, JavaScript, PHP и MySQL. Джентльменский набор Web-мастера (+ CD-ROM) / Николай Прохоренок. - М.: БХВ-Петербург, 2012. - 912 с.

27 Алиев, И. И. Экономика труда: учебник для бакалавров / И. И. Алиев, Н. А. Горелов, Л. О. Ильина. - М.: Юрайт, 2013. - 671 с. - (Серия: Бакалавр. Базовый курс)

## Приложение А Техническое задание

### Техническое задание для разработки автоматизированной системы заказа и продаж продуктов питания для ресторанов и поставщиков города Алматы

Общие требования:

- наименование разрабатываемой системы:
  - автоматизированная система заказа и продаж продуктов питания для ресторанов и поставщиков города Алматы.
- цель разработки:
  - реализовать программу для управления рисками бизнес процессов компании;
  - минимизировать риски и создать маркетплейс для оптовых продаж.
- предлагаемые технологии для разработки программы (на выбор разработчика):
  - Node.js;
  - Django;
  - jQuery;
  - bootstrap;
  - PHPMyAdmin;
  - Angular;
  - Plesk.
- выбор архитектуры построения:
  - клиент-Сервер.
- предлагаемые языки и технологии программирования:
  - Android;
  - PHP;
  - Java;
  - HTML;
  - CSS;
  - Javascript;
  - C#.
- общий объем программной части системы, Мб
  - не более 1000 Мб.

Технические требования:

- требования к программному обеспечению:
  - быстрая скорость обработки входящей информации;
  - Веб-приложение;
  - код программы должен быть читабельным и понятным.
- тестирование и отладка программного обеспечения:



- проверка работоспособности программы;

*Продолжение приложения А*

- тестирование системы продаж на корректность;
- тестирование платформы на появление ошибок и перегруженности;

Экономические требования:

- расчет стоимости системы и стоимости разработки программного обеспечения (подлежит обсуждению):
  - стоимость готового продукта 2 500 000тг;
  - стоимость разработки 1 800 000тг.
- потенциальные клиенты и области применения:
  - рестораны и места общественного питания Республики Казахстан;

## Приложение Б Акт внедрения

ТОО «RESTOFOOD(РЕСТОФУД)»  
БИН 200240017776  
Республика Казахстан,  
г. Алматы, ул.Алатау 45  
Тел. +7(705)220-42-03  
e-mail: [info@restofood.kz](mailto:info@restofood.kz)  
АО "Евразийский банк"  
KZ2994806KZT22035829  
БИК EURIKZKA

*Resto Food*

«RESTOFOOD(РЕСТОФУД)» ЖШС  
БСН 200240017776  
Қазақстан Республикасы, Алматы қ.,  
Алатау үй 45.  
Тел. +7(705)220-42-03  
e-mail: [info@restofood.kz](mailto:info@restofood.kz)  
«Еуразиялық банк» АҚ  
KZ2994806KZT22035829  
БИК EURIKZKA

Утверждаю  
Директор ТОО  
«RESTOFOOD(РЕСТОФУД)»  
Нурлан Е.Н.  
«15» мая 2020г.

### АКТ ВНЕДРЕНИЯ

Настоящий акт составлен о том, что результаты выпускной работы студента НАО «АЛМАТИНСКИЙ УНИВЕРСИТЕТ ЭНЕРГЕТИКИ И СВЯЗИ ИМЕНИ ГУМАРБЕКА ДАУКЕЕВА» группы ВТ-16-2 очной формы обучения Нурлан Е.Н. на тему «Разработка информационной системы непрерывного управления рисками для системы поддержки бизнеса компании» внедрены в ТОО «RESTOFOOD(РЕСТОФУД)» и используются в компании для закупок в сфере общественного питания. Использование результата выпускной работы Нурлан Е.Н. обеспечивает доступ к управлению рисками, создает дополнительный канал продаж, а также повышает эффективность и класс предприятия.

Директор  
ТОО «RESTOFOOD(РЕСТОФУД)»



*Нурлан* - Нурлан Е.Н.