

МИНИСТЕРСТВО ОБРАЗОВАНИЯ СВЕРДЛОВСКОЙ ОБЛАСТИ
ГОСУДАРСТВЕННОЕ АВТОНОМНОЕ ПРОФЕССИОНАЛЬНОЕ
ОБРАЗОВАТЕЛЬНОЕ УЧРЕЖДЕНИЕ СВЕРДЛОВСКОЙ ОБЛАСТИ
«КАМЕНСК-УРАЛЬСКИЙ ПОЛИТЕХНИЧЕСКИЙ КОЛЛЕДЖ»
(ГАПОУ СО «КУПК»)

Специальность 09.02.07
Информационные системы и
программирование

РАЗРАБОТКА САЙТА ДЛЯ ДЕТСКОГО
ОЗДОРОВИТЕЛЬНОГО ЛАГЕРЯ АО «ЧЭМК»

Пояснительная записка
ДП.09.02.07.01.401.275.25.ПЗ
Дипломный проект

Выполнил:
студент

_____ Боровинских П.А.

Руководитель проекта:

_____ Бодня Н.В..

Заведующий отделением
_____ Ганжа И.Н.

Консультант по экономической
части

_____ Польщикова Л.А.

СОДЕРЖАНИЕ

ВВЕДЕНИЕ	3
1 АНАЛИТИЧЕСКАЯ ЧАСТЬ	4
1.1 Описание предметной области.....	4
1.2 Описание организационной структуры предприятия	4
1.3 Описание объекта автоматизации	5
1.4 Построение модели «Как есть» для основного бизнес-процесса объекта автоматизации	6
1.5 Постановка задачи	9
2 ПРОЕКТНАЯ ЧАСТЬ.....	11
2.1 Построение модели «Как будет».....	11
2.2 Обоснование технологических решений.....	12
2.3 Проектирование и разработка серверной части программного модуля	14
2.4 Проектирование и разработка клиентской части программного модуля	34
2.5 Алгоритм внедрения программного модуля в бизнес-процесс.....	44
2.6 Условия эксплуатации.....	46
3 РАСЧЕТ СТОИМОСТИ РАЗРАБОТКИ ПРОГРАММНОГО МОДУЛЯ	48
4 ОРГАНИЗАЦИЯ РАБОЧЕГО МЕСТА И ОХРАНА ТРУДА.....	59
ЗАКЛЮЧЕНИЕ	61
СПИСОК ИСПОЛЬЗОВАННЫХ ИСТОЧНИКОВ	62
ПРИЛОЖЕНИЕ А. ТЕХНИЧЕСКОЕ ЗАДАНИЕ	63
ПРИЛОЖЕНИЕ Б. ОРГАНИЗАЦИОННАЯ СТРУКТУРА ПРЕДПРИЯТИЯ	64

					ДП.09.02.07.01.401.275.25.ПЗ			
Изм.	Лист	№ докум.	Подпись	Дата				
Разраб.		Боровинских П.А.			Разработка сайта для детского оздоровительного лагеря АО «ЧЭМК» Пояснительная записка	Лит.	Лист	Листов
Провер.		Бодня Н.В.					2	66
Реценз.		Платонов В.И.				ГАПОУ СО «КУПК» гр. ИСиП ₂₁ -401		
Н. Контр.		Лихачева Н.М.						
Утверд.		Ганжа И.Н.						

ВВЕДЕНИЕ

Современные информационные технологии играют ключевую роль в оптимизации бизнес-процессов, повышении эффективности взаимодействия с клиентами и улучшении качества предоставляемых услуг. Одной из важных сфер, где цифровизация приобретает особую значимость, является организация детского отдыха. Детские оздоровительные лагеря, такие как «Акакуль», сталкиваются с необходимостью автоматизации процессов бронирования путевок. Внедрение специализированных онлайн-инструментов позволяет упростить процедуру бронирования, сократить временные затраты сотрудников и повысить уровень удовлетворенности родителей, выбирающих лагерь для отдыха своих детей.

Актуальность данной работы обусловлена растущим спросом на удобные и интуитивно понятные системы онлайн-бронирования. Многие организации, включая детские лагеря, до сих пор используют устаревшие методы обработки заявок, такие как телефонные звонки. Это приводит к увеличению нагрузки на административный персонал, возможным ошибкам при обработке данных и замедлению процесса подтверждения брони.

Целью данного дипломного проекта является разработка и внедрение модуля онлайн-бронирования для детского оздоровительного лагеря «Акакуль».

Для достижения поставленной цели необходимо решить следующие задачи:

- Провести анализ существующих решений для онлайн-бронирования в сфере детского отдыха и туризма.
- Определить функциональные требования к модулю, исходя из потребностей лагеря и его клиентов.
- Реализовать пользовательский интерфейс для удобного взаимодействия родителей с системой бронирования.
- Реализовать функциональную часть модуля.
- Протестировать работоспособность модуля.

Практическая значимость проекта заключается в том, что внедрение модуля онлайн-бронирования позволит ДОЛ «Акакуль» оптимизировать работу с клиентами. Кроме того, система повысит конкурентоспособность лагеря за счет удобного и современного сервиса бронирования.

1 АНАЛИТИЧЕСКАЯ ЧАСТЬ

1.1 Описание предметной области

АО «Челябинский электрометаллургический комбинат» (ЧЭМК) – одно из ведущих промышленных предприятий России, специализирующееся на производстве ферросплавов и высококачественных сталей. Комбинат не только обеспечивает стабильное экономическое развитие региона, но и уделяет большое внимание социальной политике, создавая комфортные условия для труда и отдыха своих работников и их детей.

Современный ритм жизни требует не только высокой производительности труда, но и заботы о физическом и эмоциональном восстановлении. Одним из ключевых направлений в этом вопросе является организация качественного отдыха, особенно для детей сотрудников предприятия. АО "ЧЭМК" уделяет особое внимание этому аспекту, предлагая оздоровительные и развивающие программы в детском оздоровительном лагере (ДОЛ) «Акакуль».

1.2 Описание организационной структуры предприятия

Организационная структура предприятия представляет собой систему взаимосвязей между различными уровнями управления и подразделениями, обеспечивающую эффективное функционирование всего производственного комплекса. Она определяет распределение полномочий, ответственности и принципы взаимодействия между сотрудниками и отделами. В современной практике управления выделяют несколько основных типов организационных структур, каждый из которых обладает своими особенностями и применяется в зависимости от масштабов, специфики деятельности и стратегических целей предприятия.

Основные виды организационных структур:

Линейная структура управления предполагает четкое подчинение всех сотрудников единому руководителю, что обеспечивает простоту контроля и оперативность принятия решений. Однако такая модель эффективна лишь для небольших предприятий с ограниченным ассортиментом продукции, поскольку при расширении масштабов деятельности возникает перегрузка управленческого персонала.

Функциональная структура основана на разделении управления по направлениям деятельности, где каждое подразделение отвечает за определенную функцию (производство, финансы, маркетинг и т. д.). Это позволяет привлекать узких специалистов, повышая качество решений, но

					ДП.09.02.07.01.401.275.25.ПЗ	лист
Изм.	Лист	№ докум.	Подпись	Дата		4

может приводить к дублированию полномочий и снижению координации между отделами.

Линейно-функциональная структура сочетает преимущества линейного и функционального подходов. В ней линейные руководители осуществляют общее руководство и координацию, а функциональные подразделения обеспечивают экспертную поддержку по специализированным вопросам. Такая модель широко применяется на средних и крупных промышленных предприятиях, где требуется баланс между централизованным управлением и профессиональным разделением труда.

Дивизиональная структура предполагает деление предприятия на автономные подразделения (дивизионы), каждое из которых отвечает за определенный продукт, регион или направление бизнеса. Это позволяет гибко адаптироваться к изменениям рынка, но требует значительных затрат на управление.

На АО «Челябинский электрометаллургический комбинат» (ЧЭМК) реализована линейно-функциональная модель управления, которая оптимально соответствует масштабам и специфике деятельности крупного промышленного предприятия. В такой структуре линейные руководители обеспечивают контроль за основными производственными процессами и координацию работы всех подразделений, а функциональные службы сосредоточены на решении узкоспециализированных задач, таких как планирование, бухгалтерский учет, техническое развитие и логистика.

Подобная система позволяет сочетать оперативность управления с высокой компетентностью в решении профессиональных вопросов, что особенно важно для металлургического производства, требующего четкой организации и глубокой экспертизы.

Организационная структура предприятия представлена в приложении Б.

1.3 Описание объекта автоматизации

Объект автоматизации – это процесс, система или операция, которые переводятся на автоматическое управление с помощью технических средств (например, программного обеспечения, роботов, датчиков и других технологий) для повышения эффективности, точности, скорости или снижения затрат.

Объектом автоматизации в данном дипломном проекте выступает процесс продажи путевок в детский оздоровительный лагерь «Акакуль»,

					ДП.09.02.07.01.401.275.25.ПЗ	лист
Изм.	Лист	№ докум.	Подпись	Дата		5

принадлежащий АО «ЧЭМК». В современных условиях цифровизации услуг особую актуальность приобретает автоматизация процессов бронирования и оплаты, что позволяет существенно повысить эффективность работы учреждения и удобство для клиентов.

В настоящее время процесс продажи путевок организован через официальный сайт лагеря. Пользователь, желающий приобрести путевку, сначала выбирает подходящую смену из представленных вариантов. Затем указывает вид субсидии - без субсидии или муниципальная. После выбора этих параметров система перенаправляет клиента на страницу оформления заказа.

На этапе оформления заказа от пользователя требуется заполнение обязательных данных. Это полные ФИО родителя(законного представителя ребенка) и ФИО ребенка. Дополнительно запрашивается контактный телефон, адрес электронной почты(email). Также предусмотрено поле для примечаний, где можно указать особые пожелания или важную информацию о ребенке.

После успешного заполнения всех обязательных полей формы пользователь переходит к завершающему этапу - онлайн-оплате выбранной путевки.

1.4 Построение модели «Как есть» для основного бизнес-процесса объекта автоматизации

Бизнес-процесс – это логически завершенная цепочка взаимосвязанных действий, операций или задач, выполняемых в определенной последовательности для достижения конкретного бизнес-результата.

Ключевые характеристики бизнес-процесса:

Целенаправленность – каждый процесс ориентирован на достижение четко определенного результата (например, продажа товара, производство продукции, обработка заявки клиента).

Повторяемость – процесс выполняется неоднократно по стандартному алгоритму, а не как разовая операция.

Структурированность – включает последовательность шагов, участников, используемые ресурсы и правила выполнения.

Измеримость – эффективность процесса можно оценить через показатели (время выполнения, стоимость, качество).

Взаимосвязь с другими процессами – редко существует изолированно, обычно является частью более крупной бизнес-системы.

					ДП.09.02.07.01.401.275.25.ПЗ	лист
Изм.	Лист	№ докум.	Подпись	Дата		6

Для анализа текущего состояния процесса продаж путёвок в детский оздоровительный лагерь «Акакуль» было выполнено бизнес-моделирование в методологии IDEF0. IDEF0 – это методология функционального моделирования, предназначенная для структурированного описания бизнес-процессов. Она основана на графическом представлении процессов в виде блоков (функций) и стрелок (входов, выходов, управлений и механизмов), что позволяет анализировать их взаимодействие и эффективность. Этот подход позволяет системно описать все этапы работы с клиентами – от первичного обращения до заключения договора. Моделирование проводилось с целью выявления узких мест в процессе и определения направлений оптимизации продаж.

В рамках исследования была разработана модель AS-IS ("как есть"), которая отражает фактическое состояние процесса на момент анализа. Изучение текущей схемы работы показало, что существующая система продажи путевок имеет ряд особенностей, связанных со спецификой детского отдыха. Многие родители не могут сразу принять решение о покупке путевки из-за уточнения финансовых возможностей или ожидания подтверждения льготных условий. Однако текущий процесс не предусматривает гибких механизмов бронирования или поэтапной оплаты, что приводит к потере части потенциальных клиентов.

Для визуализации бизнес-процесса использовалось специализированное программное обеспечение Microsoft Visio, которое предоставляет широкие возможности для создания детализированных диаграмм. Применение методологии IDEF0 позволило четко отобразить последовательность действий, зоны ответственности сотрудников и точки взаимодействия с клиентами. Интуитивно понятный интерфейс программы обеспечил наглядное представление всех этапов процесса, что особенно важно для выявления узких мест и разработки предложений по оптимизации.

Проведенный анализ показал, что внедрение системы предварительного бронирования и гибких условий оплаты могло бы значительно повысить удобство для клиентов и увеличить количество успешных сделок. Особенно это актуально для семей, которым требуется время для окончательного подтверждения участия ребенка в смене, но которые хотели бы заранее закрепить за собой место в лагере. Такие изменения позволяют сделать процесс продажи путевок более клиентоориентированным и эффективным.

					ДП.09.02.07.01.401.275.25.ПЗ	лист
Изм.	Лист	№ докум.	Подпись	Дата		7



Рисунок 1 – Модель «Как есть» процесса продажи путёвки в детский лагерь

Входными данными в данной модели являются: данные пользователя(ФИО родителя(законного представителя), ФИО ребёнка, контактный телефон, email(электронная почта), данные банковской карты для оплаты путёвки).

Основные механизмы: сотрудники и сайт.

Компоненты управления: нормативные документы, правила лагеря и сроки оплаты.

Выходные данные: путёвка в детский оздоровительный лагерь «Акакуль».

Для более глубокого изучения существующих процессов была построена диаграмма декомпозиции модели «как есть», что даёт возможность разложить бизнес-процесс на составные элементы и проанализировать его на различных уровнях детализации.

Диаграмма изображена на рисунке 2.

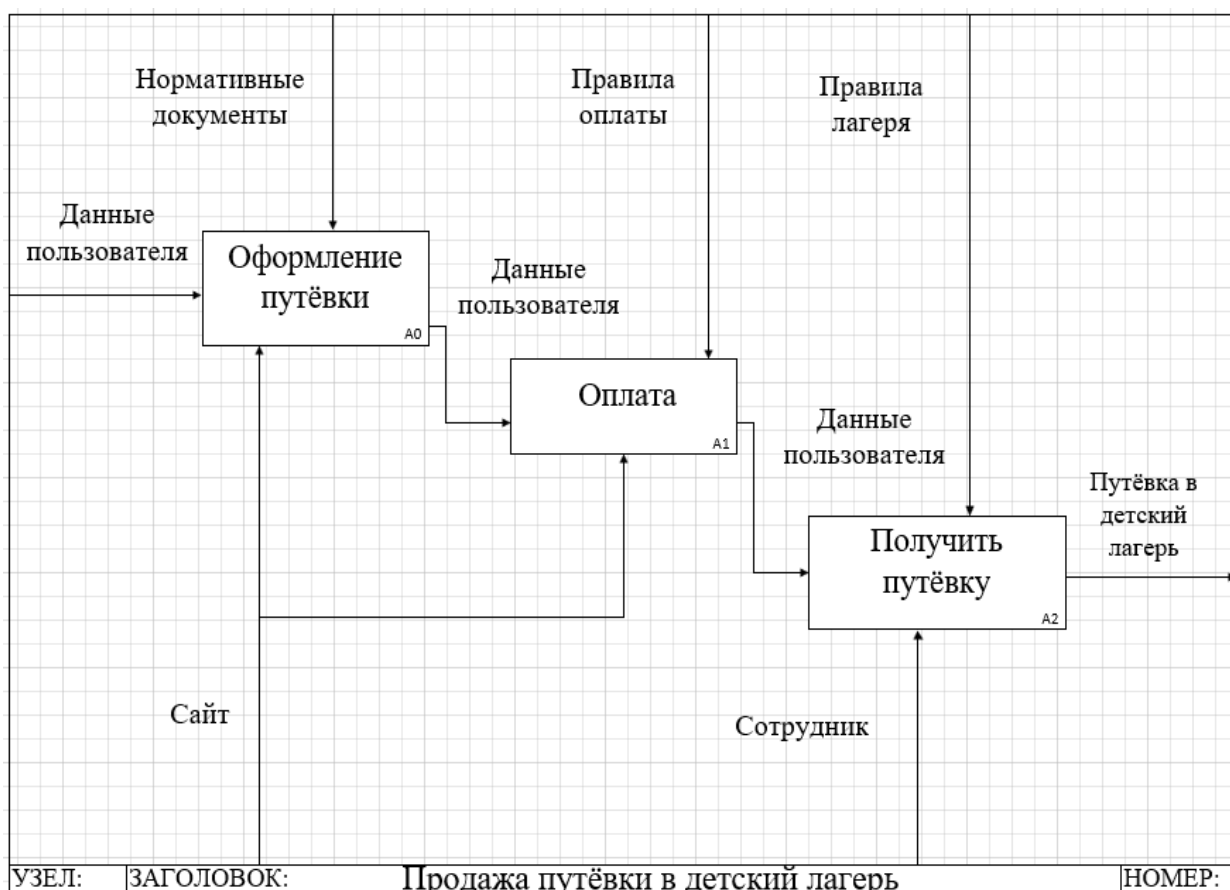


Рисунок 2 – Диаграмма декомпозиции модели «Как есть».

Входными данными в данной модели являются: данные пользователя(ФИО родителя(законного представителя), ФИО ребёнка, контактный телефон, email(электронная почта), данные банковской карты для оплаты путёвки).

Основные механизмы: сотрудник и сайт.

Компоненты управления: нормативные документы, правила лагеря и сроки оплаты.

Выходные данные: путёвка в детский оздоровительный лагерь «Акакуль».

1.5 Постановка задачи

В данном дипломном проекте поставлена задача разработать модуль онлайн-бронирования путевок для ДОЛ «Акакуль» на базе существующего сайта.

Разработка модуля онлайн-бронирования путевок для ДОЛ «Акакуль» требует решения ряда ключевых задач. Существующая система обладает существенными ограничениями, включая отсутствие механизма предварительного резервирования. Для устранения данного недостатка

необходим: провести анализ существующих решений для онлайн-бронирования в сфере детского отдыха и туризма.

- Определить функциональные требования к плагину, исходя из потребностей лагеря и его клиентов.
- Построение структуры приложения.
- Разработка ТЗ.
- Реализовать пользовательский интерфейс для удобного взаимодействия родителей с системой бронирования.
- Реализовать функциональную часть плагина.

Модуль для онлайн-бронирования путевок в детский лагерь «Акакуль» обладает рядом ключевых преимуществ, которые делают его удобным и эффективным инструментом для работы с клиентами.

Для клиентов предусмотрен удобный процесс бронирования с валидацией данных: проверка ФИО (только кириллица), телефона (формат +7 (XXX) XXX XX XX), email, дата рождения. После заполнения формы клиент получает письмо с подтверждающей ссылкой, что повышает надежность бронирования и снижает количество ошибок.

Административная панель позволяет управлять бронированиями: подтверждать, отменять, фильтровать записи по датам, ФИО, товарам и статусам. Также есть возможность ручного создания бронирований для администраторов с теми же проверками данных.

Автоматизированные уведомления по email информируют клиентов о подтверждении брони, необходимости оплаты (срок – 72 часа) или отмене (если клиент не выполнил условия). Это сокращает ручную работу и улучшает коммуникацию.

Для более эффективного анализа данных реализован экспорт бронирований в Excel с фильтрацией, а также инструмент для выполнения SQL-запросов для администраторов.

Система автоматически отменяет неоплаченные брони через 72 часа и неподтвержденные – через 24 часа, освобождая места и возвращая товары на склад.

Гибкость модуля позволяет адаптировать его не только для лагеря «Акакуль», но и для других услуг с похожей логикой бронирования, что повышает его выгодность и практичность.

Технологическая реализация будет основана на базе таких веб-технологий как: CMS WordPress, HTML, PHP, JS.

					ДП.09.02.07.01.401.275.25.ПЗ	лист
Изм.	Лист	№ докум.	Подпись	Дата		10

2 ПРОЕКТНАЯ ЧАСТЬ

2.1 Построение модели «Как будет»

Диаграмма модели ТО-ВЕ («как будет») представляет собой графическое отображение улучшенных и оптимизированных бизнес-процессов организации. Она демонстрирует целевое состояние процессов после внедрения изменений, направленных на повышение эффективности, устранение узких мест и сокращение временных затрат. Эта диаграмма служит важным инструментом коммуникации, помогая участникам процесса четко понимать новые роли, последовательность действий и способы взаимодействия.

Построение модели «как будет» осуществляется на основе анализа текущей ситуации (модели «как есть») и учитывает рекомендации по автоматизации, стандартизации и улучшению качества работы. Диаграмма «как будет» выступает в качестве дорожной карты для внедрения изменений, обеспечивая системный подход к переходу на новые практики и контроль достижения поставленных целей. В результате её применения организация не только повышает производительность, но и закладывает прочную основу для дальнейшего развития.

В новой версии бизнес-процесса был добавлен модуль для онлайн-бронирования, связанный с главным бизнес-процессом, таким как - оплата путёвки. Это решение значительно упростило и ускорило доступ к информации о путёвках, а также сделало процесс их оформления более удобным для пользователей.

Новая диаграмма выступает в качестве диаграммы декомпозиции. Диаграмма декомпозиции – это инструмент структурного анализа, который позволяет разбить сложный процесс или систему на более мелкие, управляемые компоненты. Она показывает, как высокоуровневые функции или задачи разделяются на подпроцессы, что помогает детализировать и лучше понять их структуру и взаимосвязи. Подробнее ознакомиться и увидеть основные компоненты можно на рисунке 3.

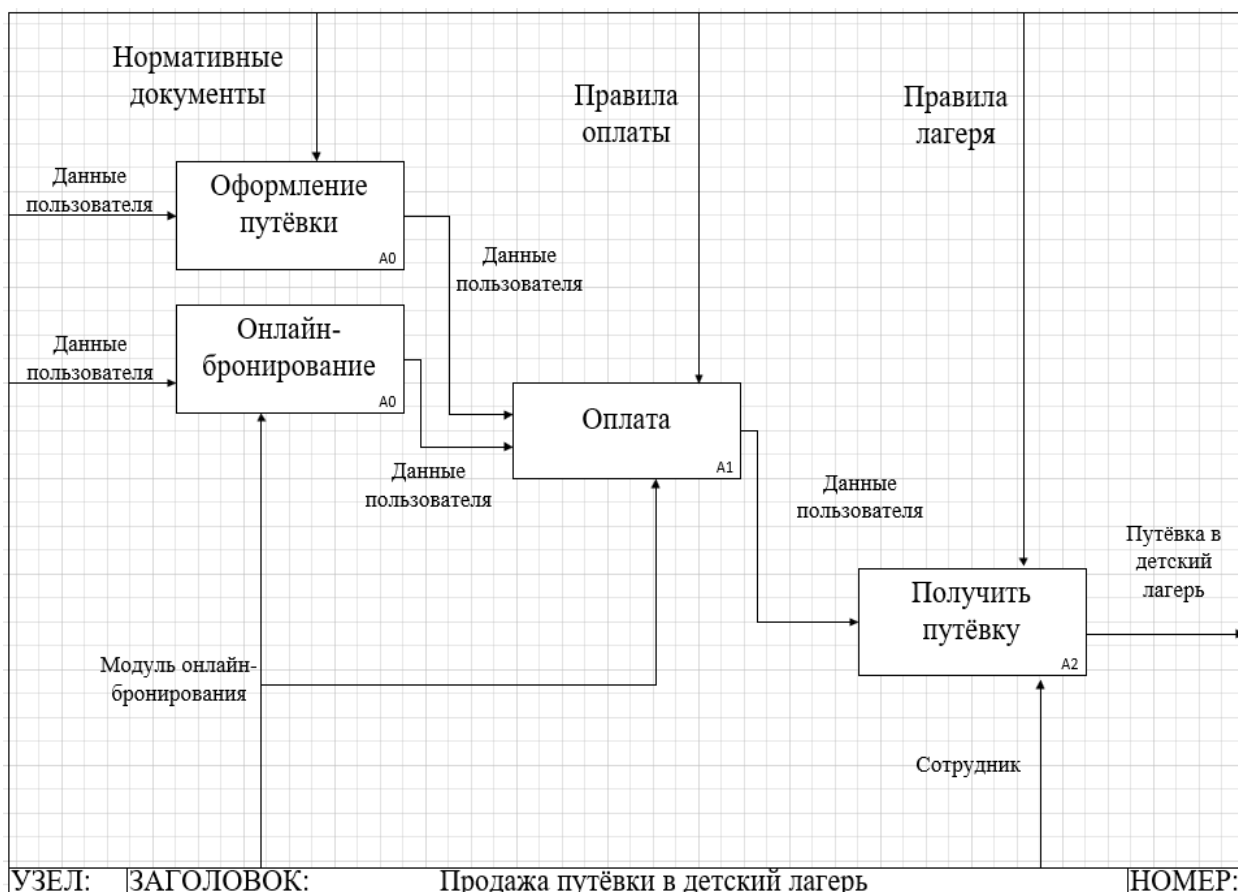


Рисунок 3 – Диаграмма «Как будет» процесса продажи путёвки в детский лагерь.

Входными данными являются: данные пользователя (ФИО родителя или законного представителя, ФИО ребёнка, дата рождения, телефон, email).

Выходные данные: путевка в ДОЛ «Акакуль».

Компоненты управления: нормативные документы, правила лагеря, сроки оплаты.

Механизмы: модуль онлайн-бронирования и сотрудник.

2.2 Обоснование технологических решений

Разработка модуля онлайн-бронирования для АО «ЧЭМК» осуществлялась с использованием проверенных технологий и инструментов, обеспечивающих надежность и простоту интеграции с существующей инфраструктурой WordPress и WooCommerce.

WordPress – свободно распространяемая система управления содержимым сайта с открытым исходным кодом; написана на PHP; сервер базы данных – MySQL. Сфера применения – от блогов до достаточно сложных новостных ресурсов. Встроенная система «тем» и «плагинов»

вместе с удачной архитектурой позволяет конструировать проекты широкой функциональной сложности.

WooCommerce – это плагин для создания интернет-магазина на базе WordPress. Он позволяет добавлять товары, обрабатывать заказы, управлять запасами и проводить платежи.

Microsoft Visio был применен для визуального проектирования бизнес-процессов, связанных с системой бронирования и покупкой путевок. Этот инструмент позволил создать четкие схемы взаимодействия между компонентами системы, что упростило процесс разработки и согласования функциональных требований. Использование Visio особенно ценно при работе со сложными процессами бронирования, где важно визуализировать все этапы - от создания заявки до подтверждения и оплаты.

Visual Studio Code стал основной средой разработки благодаря своей универсальности и богатому набору функций для работы с РНР. Редактор обеспечил комфортные условия для написания кода, включая подсветку синтаксиса, интеллектуальное завершение кода и встроенные инструменты отладки. Но также он поддерживает множество других языков программирования и обладает обширным набором расширений, что позволяет разработчикам настраивать рабочее пространство под свои нужды. Использование VS Code обеспечивает гибкость в разработке и тестировании. Это программное обеспечение также подходит для работы в командах, позволяя коллективу осуществлять совместную разработку, что особенно важно для успешной интеграции различных систем и модулей.

Модуль реализован на РНР с использованием стандартных возможностей WordPress и WooCommerce. Такой подход обеспечил несколько ключевых преимуществ:

- Полная совместимость с существующей платформой WordPress, на которой работает сайт компании.
- Использование проверенных механизмов WooCommerce для работы с товарами и заказами.
- Простота поддержки и дальнейшего развития системы.
- Надежность и безопасность, обеспечиваемые стандартными функциями WordPress.

Особое внимание уделено безопасности системы. Все критические операции защищены nonce-токенами WordPress, реализована проверка прав доступа.

Таким образом, выбранный технологический стек, основанный на PHP и стандартных возможностях WordPress/WooCommerce, обеспечивает оптимальное сочетание надежности, производительности и простоты интеграции с существующей инфраструктурой действующего сайта детского оздоровительного лагеря АО «ЧЭМК».

2.3 Проектирование и разработка серверной части программного модуля

Серверная часть системы реализована с использованием стандартных механизмов WordPress - шорткодов и хуков, что обеспечивает простоту интеграции с любыми темами. Основной язык написания модуля- php, для валидации форм использовался такой язык, как JavaScript, что повышает удобство для заполнения форм и обработки данных. Шорткод в WordPress – это специальная короткая команда в квадратных скобках, которая заменяется на более сложный код или функционал при загрузке страницы.

Хуки в WordPress – это специальные "крючки", которые позволяют добавлять свой код или изменять стандартное поведение сайта без редактирования исходных файлов WordPress.

Akakul-booking.php- ядро модуля. Проверяет наличие WooCommerce, подключение стилей и скриптов, создание таблицы базы данных для хранения бронирования и активации модуля, удаление таблицы при удалении модуля, запуск основных классов(Akakul_Booking_Frontend, Akakul_Booking_Admin, Akakul_Booking_Manual).

На рисунке 4 происходит создание и удаление таблицы при активации модуля. При активации плагина создается таблица woocommerce_bookings с полями для хранения данных о бронированиях (ID товара, ФИО, email, даты и т.д.). При удалении плагина эта таблица автоматически удаляется

```

register_activation_hook(__FILE__, 'create_booking_table');

function create_booking_table() {
    global $wpdb;
    $table_name = $wpdb->prefix . 'woocommerce_bookings';
    $wpdb->query("DROP TABLE IF EXISTS $table_name");
    $charset_collate = $wpdb->get_charset_collate();

    // В функции create_booking_table() обновим SQL-запрос:
    $sql = "CREATE TABLE $table_name (
        id mediumint(9) NOT NULL AUTO_INCREMENT,
        product_id mediumint(9) NOT NULL,
        product_name varchar(255) NOT NULL,
        parent_name varchar(255) NOT NULL,
        child_name varchar(255) NOT NULL,
        email varchar(255) NOT NULL,
        phone varchar(50) NOT NULL,
        child_bdate varchar(50) NOT NULL,
        booking_date datetime DEFAULT CURRENT_TIMESTAMP NOT NULL,
        confirmation_token varchar(64) DEFAULT NULL,
        is_confirmed tinyint(1) DEFAULT 0,
        order_id bigint(20) DEFAULT NULL,
        confirm_deadline int(11) DEFAULT NULL,
        payment_deadline int(11) DEFAULT NULL,
        PRIMARY KEY (id)
    ) $charset_collate;";

    require_once(ABSPATH . 'wp-admin/includes/upgrade.php');
    dbDelta($sql);
}

// Удаляем таблицу при удалении плагина
register_uninstall_hook(__FILE__, 'delete_booking_table');

function delete_booking_table() {
    global $wpdb;
    $table_name = $wpdb->prefix . 'woocommerce_bookings';

    // Удаляем таблицу, если она существует
    $wpdb->query("DROP TABLE IF EXISTS $table_name");
}

```

Рисунок 4 – Создание и удаление таблицы при активации модуля.

На рисунке 5 представлено подключение стилей, благодаря которым можно изменять внешний вид веб-страницы.

					ДП.09.02.07.01.401.275.25.ПЗ	ЛИСТ
Изм.	Лист	№ докум.	Подпись	Дата		15

```

function akakul_booking_admin_enqueue_styles() {
    wp_register_style(
        'akakul-booking-admin-style',
        plugins_url('assets/css/admin-style.css', __FILE__),
        array(),
        filemtime(plugin_dir_path(__FILE__) . 'assets/css/admin-style.css')
    );

    wp_enqueue_style('akakul-booking-admin-style');
}
// frontend
function akakul_booking_frontend_enqueue_styles() {
    wp_register_style(
        'akakul-booking-frontend-style',
        plugins_url('assets/css/frontend-style.css', __FILE__),
        array(),
        filemtime(plugin_dir_path(__FILE__) . 'assets/css/frontend-style.css')
    );

    wp_enqueue_style('akakul-booking-frontend-style');
}
add_action('admin_enqueue_scripts', 'akakul_booking_admin_enqueue_styles');

```

Рисунок 5 – Подключение файлов стилей.

На рисунке 6 представлено подключение скриптов, благодаря которым можно применять какие-либо интерактивные действия, например, маска для поля «телефон».


```

function akakul_booking_enqueue_scripts() {
    wp_enqueue_script(
        'akakul-booking-frontend-script',
        plugins_url('assets/js/frontend-script.js', __FILE__),
        array('jquery'),
        filemtime(plugin_dir_path(__FILE__) . 'assets/js/frontend-script.js'),
        true
    );

    // Подключаем библиотеку jQuery Mask Plugin
    wp_enqueue_script(
        'jquery-mask-plugin',
        plugins_url('assets/js/jquery.mask.min.js', __FILE__),
        array('jquery'),
        '1.14.16',
        true
    );
}
add_action('wp_enqueue_scripts', 'akakul_booking_enqueue_scripts');

```

Рисунок 6 – Подключение скриптов.

На рисунке 7 изображена функция, в которой формируется задачи для Cron (служит для выполнения запланированных задач в определённое время, дату или через интервал), а именно это задачи по удалению бронирований, если они не подтверждены в течение 24 часов, и не оплачены в течение 72 часов.

```

// Обработчик задачи Cron
add_action('akakul_booking_check_unconfirmed_bookings', 'akakul_booking_check_unconfirmed_bookings_handler');

function akakul_booking_check_unconfirmed_bookings_handler() {
    global $wpdb;
    $table_name = $wpdb->prefix . 'woocommerce_bookings';
    $now = time();

    // удаление неподтвержденные брони старше 24 часов
    $unconfirmed_bookings = $wpdb->get_results($wpdb->prepare(
        "SELECT * FROM $table_name
        WHERE is_confirmed = 0 AND confirm_deadline < %d",
        $now
    ));

    foreach ($unconfirmed_bookings as $booking) {
        // отправка уведомления
        $subject = __('Ваше бронирование истекло', 'woocommerce-booking');
        $message = sprintf(
            __('Здравствуйте, %s!<br><br>Ваше бронирование товара "%s" было автоматически отменено,
            так как вы не подтвердили его в течение 24 часов.', 'woocommerce-booking'),
            $booking->parent_name,
            $booking->product_name
        );

        add_filter('wp_mail_content_type', function() { return 'text/html'; });
        wp_mail($booking->email, $subject, $message);
        remove_filter('wp_mail_content_type', 'set_html_content_type');

        // удаление записи
        $wpdb->delete($table_name, array('id' => $booking->id));
    }
}

```

Рисунок 7 – Обработчик событий Cron.

На рисунке 8 подключаются основные классы, которые используются при разработке модуля.

```

require('includes/class-akakul-booking-frontend.php');
include('includes/class-akakul-booking-admin.php');
include('includes/class-akakul-booking-manual.php');

```

Рисунок 8 – Подключение основных классов.

Перейдем к классу - class-akakul-booking-admin, в котором находятся функции, которые отвечают за работу админки(отображение, фильтрация, управление бронированиями и выполнение SQL-запросов).

```
class Akakul_Booking_Admin {
// конструктор
public function __construct() {
    add_action('admin_menu', array($this, 'add_booking_admin_page'));
    add_action('admin_menu', array($this, 'add_sql_query_page'));
    add_action('admin_init', array($this, 'handle_delete_booking'));
    add_action('admin_init', array($this, 'handle_export_excel'));
    add_action('admin_enqueue_scripts', array($this, 'enqueue_admin_styles'));
}
```

На рисунке 9 можно увидеть функцию __construct(), которая инициализирует класс и регистрирует хуки и шорткоды.

Рисунок 9 – Конструктор класса.

Рисунок 10 представляет собой подключение стилей.

```
// стили
public function enqueue_admin_styles() {
    wp_enqueue_style(
        'akakul-booking-admin-style',
        plugins_url('assets/css/admin-style.css', dirname(__FILE__)),
        array(),
        filemtime(plugin_dir_path(dirname(__FILE__)) . 'assets/css/admin-style.css')
    );
}
```

Рисунок 10 – Подключение стилей.

Рисунки 11, 12 содержат в себе функции для добавления меню и подменю в админ-панели.

```
// меню
public function add_booking_admin_page() {
    add_menu_page(
        ('Бронирования', 'woocommerce-booking'),
        ('Бронирования', 'woocommerce-booking'),
        'edit_posts',
        'woocommerce-bookings',
        array($this, 'render_booking_admin_page'),
        'dashicons-calendar-alt',
        6
    );
}
```

Рисунок 11 – Добавление меню.

```
// подменю
public function add_sql_query_page() {
    add_submenu_page(
        'woocommerce-bookings',
        ('Настройки', 'woocommerce-booking'),
        ('Настройки', 'woocommerce-booking'),
        'manage_options',
        'sql-query',
        array($this, 'render_sql_query_page'),
        'dashicons-admin-generic',
        10
    );
}
```

Рисунок 12 – Добавление подменю.

Рассмотрим функцию `render_booking_admin_page()` – основная страница в админке, включающая в себя снятие и подтверждение бронирования, а также фильтрацию, подробный код представлен на рисунках 13, 14, 15.

```

// основная страница
public function render_booking_admin_page() {
    global $wpdb;
    $table_name = $wpdb->prefix . 'woocommerce_bookings';

    // Обработка снятия бронирования
    if (isset($_GET['action']) && $_GET['action'] === 'unbook' && isset($_GET['id'])) {
        if (!wp_verify_nonce($_GET['_wpnonce'], 'unbook_booking_' . $_GET['id'])) {
            wp_die(__('Недействительный запрос.', 'woocommerce_booking'));
        }

        $booking_id = intval($_GET['id']);
        $booking = $wpdb->get_row($wpdb->prepare("SELECT * FROM $table_name WHERE id = %d", $booking_id));

        if ($booking) {
            // Отправляем уведомление клиенту
            $this->send_booking_cancellation_email(
                $booking,
                __('Бронирование было снято администратором', 'woocommerce_booking')
            );
        }

        $wpdb->update(
            $table_name,
            array('is_confirmed' => 0),
            array('id' => $booking_id)
        );

        wp_redirect(admin_url('admin.php?page=woocommerce-bookings'));
        exit;
    }
}

```

Рисунок 13 – Снятие бронирования.

```

// Обработка подтверждения бронирования
if (isset($_GET['action']) && $_GET['action'] === 'confirm' && isset($_GET['id'])) {
    if (!wp_verify_nonce($_GET['_wpnonce'], 'confirm_booking_' . $_GET['id'])) {
        wp_die(__('Недействительный запрос.', 'woocommerce-booking'));
    }

    $booking_id = intval($_GET['id']);
    $wpdb->update(
        $table_name,
        array('is_confirmed' => 1),
        array('id' => $booking_id)
    );

    wp_redirect(admin_url('admin.php?page=woocommerce-bookings'));
    exit;
}

```

Рисунок 14 – Подтверждение бронирования.

```

// Фильтры
$filter_product_name = isset($_GET['filter_product_name']) ? sanitize_text_field($_GET['filter_product_name']) : '';
$filter_parent_name = isset($_GET['filter_parent_name']) ? sanitize_text_field($_GET['filter_parent_name']) : '';
$filter_child_name = isset($_GET['filter_child_name']) ? sanitize_text_field($_GET['filter_child_name']) : '';
$filter_child_bdate = isset($_GET['filter_child_bdate']) ? sanitize_text_field($_GET['filter_child_bdate']) : '';
$filter_email = isset($_GET['filter_email']) ? sanitize_text_field($_GET['filter_email']) : '';
$filter_phone = isset($_GET['filter_phone']) ? sanitize_text_field($_GET['filter_phone']) : '';
$filter_booking_date = isset($_GET['filter_booking_date']) ? sanitize_text_field($_GET['filter_booking_date']) : '';
$filter_confirmed = isset($_GET['filter_confirmed']) ? sanitize_text_field($_GET['filter_confirmed']) : '';

// Формируем SQL-запрос с учетом фильтров
$where = array();
if ($filter_product_name) {
    $where[] = $wpdb->prepare("product_name LIKE %s", "%$filter_product_name%");
}
if ($filter_parent_name) {
    $where[] = $wpdb->prepare("parent_name LIKE %s", "%$filter_parent_name%");
}
if ($filter_child_name) {
    $where[] = $wpdb->prepare("child_name LIKE %s", "%$filter_child_name%");
}
if ($filter_child_bdate) {
    $where[] = $wpdb->prepare("child_bdate = %s", $filter_child_bdate);
}
if ($filter_email) {
    $where[] = $wpdb->prepare("email LIKE %s", "%$filter_email%");
}
if ($filter_phone) {
    $where[] = $wpdb->prepare("phone LIKE %s", "%$filter_phone%");
}
if ($filter_booking_date) {
    $where[] = $wpdb->prepare("DATE(booking_date) = %s", $filter_booking_date);
}
if ($filter_confirmed !== '') {
    $where[] = $wpdb->prepare("is_confirmed = %d", $filter_confirmed);
}
$where = $where ? 'WHERE ' . implode(' AND ', $where) : '';

```

Рисунок 15 – Фильтрация.

					ДП.09.02.07.01.401.275.25.ПЗ	лист
Изм.	Лист	№ докум.	Подпись	Дата		22

Функция `handle_delete_booking()` обрабатывает удаление бронирования. Включает в себя проверку на безопасность, получение данных бронирования из БД, отправку сообщений клиенту о том, что бронирование удалено по email, возвращение товара на склад и удаление записи из таблицы бронирования. Функция представлена на рисунке 16.

```
public function handle_delete_booking() {
    // проверка на безопасность при помощи nonce
    if (isset($_GET['action']) && $_GET['action'] === 'delete' && isset($_GET['id'])) {
        if (!wp_verify_nonce($_GET['_wpnonce'], 'delete_booking_' . $_GET['id'])) {
            wp_die(__('Недействительный запрос.', 'woocommerce-bookings'));
        }

        global $wpdb;
        $table_name = $wpdb->prefix . 'woocommerce_bookings';

        // Получаем данные о бронировании
        $booking = $wpdb->get_row($wpdb->prepare("SELECT * FROM $table_name WHERE id = %d", intval($_GET['id'])));

        if ($booking) {
            // Отправляем уведомление клиенту
            $this->send_booking_cancellation_email(
                $booking,
                __('Бронирование было удалено администратором', 'woocommerce-bookings')
            );

            // Возвращаем товар на склад
            $product = wc_get_product($booking->product_id);
            if ($product) {
                $current_stock = $product->get_stock_quantity();
                $product->set_stock_quantity($current_stock + 1);
                $product->save();
            }

            // Удаляем бронирование
            $wpdb->delete($table_name, array('id' => intval($_GET['id'])));
        }

        wp_redirect(admin_url('admin.php?page=woocommerce-bookings'));
        exit;
    }
}
```

Рисунок 16 – Функция удаления бронирования.

Функция `handle_export_excel()` помогает экспортировать данные бронирований в Excel-файл. Данная функция получает данные из БД, применяет те же фильтры, что и на странице бронирования, генерирует HTML таблицу, которую можно открыть в Excel. Подробнее в рисунках 17 и 18.

					ДП.09.02.07.01.401.275.25.ПЗ	лист
Изм.	Лист	№ докум.	Подпись	Дата		23

```

public function handle_export_excel() {
    if (isset($_GET['action']) && $_GET['action'] === 'export_excel') {
        global $wpdb;
        $table_name = $wpdb->prefix . 'woocommerce_bookings';

        // Получаем данные с учетом фильтров
        $where = array();
        $filters = array(
            'filter_product_name', 'filter_parent_name', 'filter_child_name',
            'filter_child_bdate', 'filter_email', 'filter_phone',
            'filter_booking_date', 'filter_confirmed'
        );

        foreach ($filters as $filter) {
            if (!empty($_GET[$filter])) {
                $value = sanitize_text_field($_GET[$filter]);
                switch ($filter) {
                    case 'filter_product_name':
                        $where[] = $wpdb->prepare("product_name LIKE %s", "%$value%");
                        break;
                    case 'filter_parent_name':
                        $where[] = $wpdb->prepare("parent_name LIKE %s", "%$value%");
                        break;
                    case 'filter_child_name':
                        $where[] = $wpdb->prepare("child_name LIKE %s", "%$value%");
                        break;
                    case 'filter_child_bdate':
                        $where[] = $wpdb->prepare("child_bdate = %s", $value);
                        break;
                    case 'filter_email':
                        $where[] = $wpdb->prepare("email LIKE %s", "%$value%");
                        break;
                    case 'filter_phone':
                        $where[] = $wpdb->prepare("phone LIKE %s", "%$value%");
                        break;
                    case 'filter_booking_date':
                        $where[] = $wpdb->prepare("DATE(booking_date) = %s", $value);
                        break;
                    case 'filter_confirmed':
                        $where[] = $wpdb->prepare("is_confirmed = %d", $value);
                        break;
                }
            }
        }

        $where_clause = $where ? 'WHERE ' . implode(' AND ', $where) : '';
        $bookings = $wpdb->get_results("SELECT * FROM $table_name $where_clause ORDER BY booking_date DESC");
    }
}

```

Рисунок 17 – Получение данных с учетом фильтров.


```

// Генерируем Excel
header('Content-Type: application/vnd.ms-excel; charset=utf-8');
header('Content-Disposition: attachment; filename=bookings_' . date('Y-m-d') . '.xls');
header('Pragma: no-cache');
header('Expires: 0');

echo "<html>";
echo "<meta http-equiv=\"Content-Type\" content=\"text/html; charset=utf-8\" />";
echo "<table border='1'>";

// Заголовки таблицы
echo "<tr>";
echo "<th>ID</th>";
echo "<th>Товар</th>";
echo "<th>Ф.И.О. родителя</th>";
echo "<th>Ф.И.О. ребенка</th>";
echo "<th>Дата рождения ребенка</th>";
echo "<th>Email</th>";
echo "<th>Телефон</th>";
echo "<th>Дата бронирования</th>";
echo "<th>Подтверждено</th>";
echo "<th>ID заказа</th>";
echo "<th>Статус заказа</th>";
echo "</tr>";

// Данные
foreach ($bookings as $booking) {
    $product = wc_get_product($booking->product_id);
    $product_name = $product ? $product->get_name() : 'Товар удален';

    $order_status = '';
    if ($booking->order_id) {
        $order = wc_get_order($booking->order_id);
        $order_status = $order ? $order->get_status() : 'Заказ не найден';
    }

    echo "<tr>";
    echo "<td>" . esc_html($booking->id) . "</td>";
    echo "<td>" . esc_html($product_name) . "</td>";
    echo "<td>" . esc_html($booking->parent_name) . "</td>";
    echo "<td>" . esc_html($booking->child_name) . "</td>";
    echo "<td>" . esc_html($booking->child_bdate) . "</td>";
    echo "<td>" . esc_html($booking->email) . "</td>";
    echo "<td>" . esc_html($booking->phone) . "</td>";
    echo "<td>" . esc_html($booking->booking_date) . "</td>";
    echo "<td>" . ($booking->is_confirmed ? 'Да' : 'Нет') . "</td>";
    echo "<td>" . ($booking->order_id ? esc_html($booking->order_id) : '') . "</td>";
    echo "<td>" . esc_html($order_status) . "</td>";
    echo "</tr>";
}

```

Рисунок 18 – Генерация данных.

Функция `send_booking_cancellation_email()` отправляет email клиенту при отмене или удалении бронирования. Код на рисунке 19.

```
// отправка сообщения клиенту об отмене
private function send_booking_cancellation_email($booking, $reason) {
    $subject = __('Ваше бронирование отменено', 'woocommerce-booking');

    $message = sprintf(
        __('Здравствуйте, %s!<br><br>Ваше бронирование товара "%s" было отменено.<br><br>
        Причина: %s<br><br>Если вы считаете, что это произошло по ошибке, пожалуйста, свяжитесь с нами.', 'woocommerce-booking'),
        $booking->parent_name,
        $booking->product_name,
        $reason
    );

    add_filter('wp_mail_content_type', function() { return 'text/html'; });
    wp_mail($booking->email, $subject, $message);
    remove_filter('wp_mail_content_type', 'set_html_content_type');
}
```

Рисунок 19 – Сообщение с причиной отмены.

Функция `render_sql_query_page()` предоставляет возможность выполнить SQL-запрос. Права доступа к этой странице есть только у администратора, соответственно при помощи данной функции можно выполнить SQL-запрос и получить, например, таблицу модуля, которая создается при активации системы (`wp_woocommerce_bookings`). Подробнее на рисунке 20.

```

public function render_sql_query_page() {
    if (!current_user_can('manage_options')) {
        wp_die(__('У вас недостаточно прав для доступа к этой странице.', 'woocommerce-booking'));
    }

    global $wpdb;
    $results = [];
    $error = '';
    $query = '';

    if (isset($_POST['sql_query'])) {
        if (!isset($_POST['sql_query_nonce']) || !wp_verify_nonce($_POST['sql_query_nonce'], 'sql_query_action')) {
            $error = 'Ошибка безопасности. Попробуйте еще раз.';
        } else {
            $query = stripslashes($_POST['sql_query']);
            if (!empty($query)) {
                try {
                    $results = $wpdb->get_results($query, ARRAY_A);
                } catch (Exception $e) {
                    $error = 'Ошибка выполнения запроса: ' . $e->getMessage();
                }
            } else {
                $error = 'SQL-запрос не может быть пустым.';
            }
        }
    }
}

```

Рисунок 20 – Выполнение запроса и проверка на право доступа.

Класс class-akakul-booking-manual – в нем находится функционал для создания бронирования через адмн-панель.

Функция __construct()- инициализирует класс и добавляет хук для создания страницы бронирования в админ-панели. Код на рисунке 21.

```

public function __construct() {
    add_action('admin_menu', array($this, 'add_admin_booking_page'));
}

```

Рисунок 21 – Инициализация класса и создание страницы бронирования.

Функция add_admin_booking_page() – добавляет подменю. Код представлен на рисунке 22.

```
// подмену
public function add_admin_booking_page() {
    add_submenu_page(
        'woocommerce-bookings', // Родительский slug (страница бронирований)
        ('Забронировать товар', 'woocommerce-booking'), // Заголовок страницы
        ('Забронировать товар', 'woocommerce-booking'), // Название в меню
        'edit_posts', // Права доступа
        'admin-booking', // Slug страницы
        array($this, 'render_admin_booking_page') // Функция для отображения страницы
    );
}
```

Рисунок 22 – Код для реализации подмену.

Функция `render_admin_booking_page()` - отображает страницу бронирования в админ-панели. В ней реализовано проверка прав доступа, обрабатывается отправка формы, есть возможность выбрать товар для бронирования. Подробнее на рисунках 23, 24, 25.

```
public function render_admin_booking_page() {
    // Проверяем права доступа
    if (!current_user_can('edit_posts')) {
        wp_die(__('У вас недостаточно прав для доступа к этой странице.', 'woocommerce-booking'));
    }
}
```

Рисунок 23 – Проверка прав доступа.

```
// Обработка отправки формы
$error_message = '';
if (isset($_POST['submit_admin_booking'])) {
    $error_message = $this->validate_admin_booking_data();
    if (empty($error_message)) {
        $result = $this->handle_admin_booking_request();
        if ($result) {
            echo '<div class="notice notice-success"><p>' . __('Бронирование успешно создано!', 'woocommerce-booking') . '</p></div>';
        } else {
            echo '<div class="notice notice-error"><p>' . __('Ошибка при создании бронирования.', 'woocommerce-booking') . '</p></div>';
        }
    } else {
        echo '<div class="notice notice-error"><p>' . esc_html($error_message) . '</p></div>';
    }
}
```

Рисунок 24 – Обработка отправки формы.

					ДП.09.02.07.01.401.275.25.ПЗ	лист
Изм.	Лист	№ докум.	Подпись	Дата		28

```
// Получаем все товары WooCommerce
$products = wc_get_products(array(
    'status' => 'publish',
    'limit' => -1,
));
```

Рисунок 25 – Получение товаров.

Функция `validate_admin_booking_data()` – проверяет корректность данных, отправленных через форму, представлена на рисунке 26.

```
private function validate_admin_booking_data() {
    if (empty($_POST['parent_name']) || empty($_POST['child_name']) ||
        empty($_POST['child_bdate']) || empty($_POST['email']) ||
        empty($_POST['phone'])) {
        return __('Все поля обязательны для заполнения.', 'woocommerce-booking');
    }

    // Валидация ФИО
    if (!preg_match('/^[А-Яа-яёЁ\s]+$/u', $_POST['parent_name']) || count(explode(' ', $_POST['parent_name'])) < 3) {
        return __('ФИО родителя должно содержать только кириллицу и быть в формате "Фамилия Имя Отчество".', 'woocommerce-booking');
    }

    if (!preg_match('/^[А-Яа-яёЁ\s]+$/u', $_POST['child_name']) || count(explode(' ', $_POST['child_name'])) < 3) {
        return __('ФИО ребенка должно содержать только кириллицу и быть в формате "Фамилия Имя Отчество".', 'woocommerce-booking');
    }

    // Валидация телефона
    if (!preg_match('/^+7 \(\d{3}\) \d{3} \d{2} \d{2}$/u', $_POST['phone'])) {
        return __('Неверный формат телефона. Используйте формат: +7 (999) 999 99 99', 'woocommerce-booking');
    }

    return '';
}
```

Рисунок 26 – Валидация формы.

Функция `handle_admin_booking_request()` – обрабатывает и сохраняет данные бронирования. Сначала проверяется товар на его доступность, уменьшение товара при брони, генерация уникального токена для подтверждения, сохранение в таблицу `wp_woocommerce_bookings` со статусом «подтверждено». Код представлен на рисунках 27, 28, 29.

					ДП.09.02.07.01.401.275.25.ПЗ	лист
Изм.	Лист	№ докум.	Подпись	Дата		29

```

private function handle_admin_booking_request() {
    global $wpdb;
    $table_name = $wpdb->prefix . 'woocommerce_bookings';

    // Проверяем, заполнены ли все поля
    if (empty($_POST['product_id']) || empty($_POST['parent_name']) || empty($_POST['child_name']) ||
        empty($_POST['child_bdate']) || empty($_POST['email']) || empty($_POST['phone'])) {
        return false;
    }

    $product_id = intval($_POST['product_id']);
    $current_product = wc_get_product($product_id);

    // Проверяем, есть ли товар в наличии
    if (!$current_product || !$current_product->is_in_stock()) {
        echo '<div class="notice notice-error"><p> . __(('Товар недоступен для бронирования.', 'woocommerce-booking') . '</p></div>';
        return false;
    }
}

```

Рисунок 27 – Проверка на заполнение и наличие товара.

```

// Уменьшаем количество товара на складе
$current_stock = $current_product->get_stock_quantity();
if ($current_stock > 0) {
    $current_product->set_stock_quantity($current_stock - 1);
    $current_product->save();
} else {
    echo '<div class="notice notice-error"><p> . __(('Товар закончился на складе.', 'woocommerce-booking') . '</p></div>';
    return false;
}

// Собираем данные из формы
$product_name = sanitize_text_field($current_product->get_name());
$parent_name = sanitize_text_field($_POST['parent_name']);
$child_name = sanitize_text_field($_POST['child_name']);
$child_bdate = sanitize_text_field($_POST['child_bdate']);
$email = sanitize_email($_POST['email']);
$phone = sanitize_text_field($_POST['phone']);

```

Рисунок 28 – Уменьшение товара на складе и сбор данных из формы.

```

// Генерация токена подтверждения
$confirmation_token = $this->generate_booking_token();

// Сохраняем бронирование в базу данных
$result = $wpdb->insert(
    $table_name,
    array(
        'product_id' => $product_id,
        'product_name' => $product_name,
        'parent_name' => $parent_name,
        'child_name' => $child_name,
        'child_bdate' => $child_bdate,
        'email' => $email,
        'phone' => $phone,
        'confirmation_token' => $confirmation_token,
        'is_confirmed' => 1, // Автоматически подтверждаем бронирование, созданное администратором
        'booking_date' => current_time('mysql'), // текущее время
    )
);

```

Рисунок 29 – Генерация токена подтверждения и сохранение бронирования в БД.

Функция `generate_booking_token()` – генерирует уникальный токен для бронирования. Токен - это случайно сгенерированная строка символов которая служит «цифровым ключом» для идентификации и подтверждения бронирования. Код на рисунке 30.

```

private function generate_booking_token() {
    return bin2hex(random_bytes(16)); // Генерация случайного токена
}

```

Рисунок 30 – Генерация случайного токена.

На рисунке 31 можно увидеть, как реализовано меню в админ-панели.

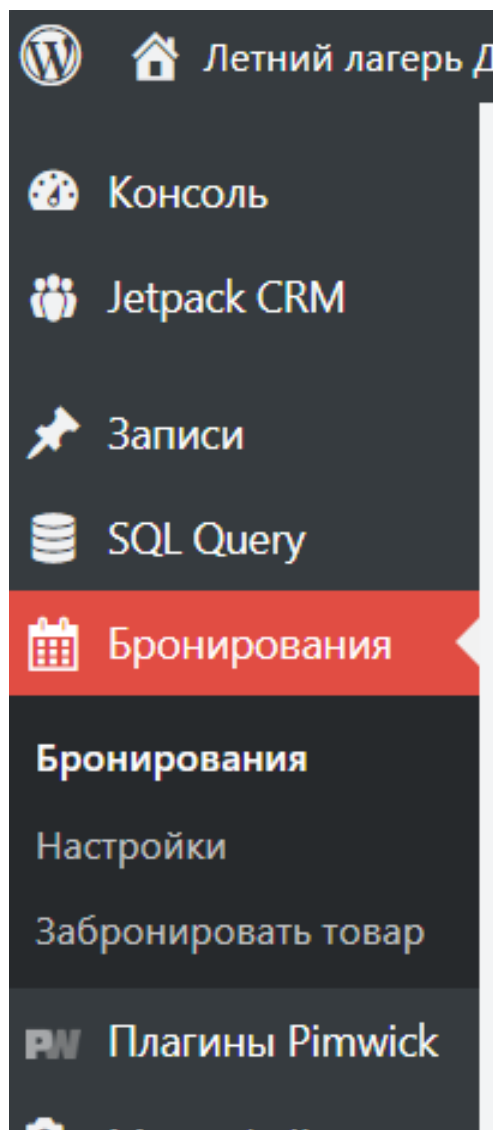


Рисунок 31 – Меню в админ-панели.

Рисунок 32 представляет собой основную страницу с выводом всех бронирований.

Бронирования

Экспорт в Excel

№	Товар	Ф.И.О. родителя	Ф.И.О. ребенка	Д.р. ребенка	Email	Телефон	Дата брони	Подтверждено	Действия
	<input type="text"/>	<input type="text"/>	<input type="text"/>	<input type="text" value="ДД.ММ.ГГГГ"/>	<input type="text"/>	<input type="text"/>	<input type="text" value="ДД.ММ.ГГГГ"/>	<input type="text" value="Все"/>	<input type="button" value="Фильтровать"/> <input type="button" value="Сбросить фильтры"/>
7	Товар для проверки оплаты	Иванов Иван Иванович	Иванов Иван Иванович	2011-01-28	borovinskikhpolina@yandex.ru	+7 (777) 777 77 77	2025-05-28 22:53:48	Да	<input type="button" value="Снять"/> <input type="button" value="Удалить"/>

Рисунок 32 – Основная страница в админ-панели.

На рисунке 33 представлена форма для бронирования, как ее видит администратор.

Забронировать товар

Выберите товар:

Ф.И.О. родителя (полностью):
Формат: Фамилия Имя Отчество

Ф.И.О. ребенка (полностью):
Формат: Фамилия Имя Отчество

Дата рождения ребенка:

Email:

Телефон:
Формат: +7 (999) 999 99 99

Забронировать

Рисунок 33 – Форма для бронирования путёвки администратором.

Рисунок 34 – как выглядит форма для SQL-запросов в админ-панели.

SQL Запросы

Выполнить запрос

Рисунок 34 – Форма для SQL-запросов.

2.4 Проектирование и разработка клиентской части программного модуля

Клиентская часть отвечает за взаимодействие с пользователями на сайте: отображение формы бронирования, сбор данных и их валидацию перед отправкой на сервер. Дизайн и стилизация форм были разработаны в едином стиле с существующим сайтом, чтобы обеспечить бесшовную интеграцию и сохранить визуальную согласованность.

Класс реализует функционал бронирования товаров со стороны пользователя, то есть с клиентской части модуля.

Функция `__construct()` - инициализирует класс и регистрирует хуки и шорткоды. Код на рисунке 35.

```
public function __construct() {  
    add_action('woocommerce_after_add_to_cart_button', array($this, 'add_booking_button'));  
    add_shortcode('woocommerce_booking_page', array($this, 'render_booking_page'));  
    add_action('init', array($this, 'handle_booking_confirmation'));  
}
```

Рисунок 35 – Инициализация класса, регистрация хуков и шорткодов.

Функция `add_booking_button()` добавляет кнопку «забронировать» на странице товара. Представлена на рисунке 36.

```
// кнопка забронировать  
public function add_booking_button() {  
    global $product;  
    $booking_url = home_url('/bronirowanie/?product_id=' . $product->get_id());  
    echo '<a href="' . esc_url($booking_url) . '" class="button" id="front_zabr" style="'  
        width: 150px;  
        height: 30px;  
        background-color: #FC7234;  
        margin-left: 10px;  
        text-decoration: none;  
        display: inline-block;  
        color: white;  
        text-align: center;  
        line-height: 30px;  
        border-radius: 5px;  
        font-size: 16px;">' . __('Забронировать', 'woocommerce-booking') . '</a>';  
}
```

Рисунок 36 – Кнопка «забронировать».

На рисунке 37 представлено, как выглядит кнопка для бронирования товара на сайте.

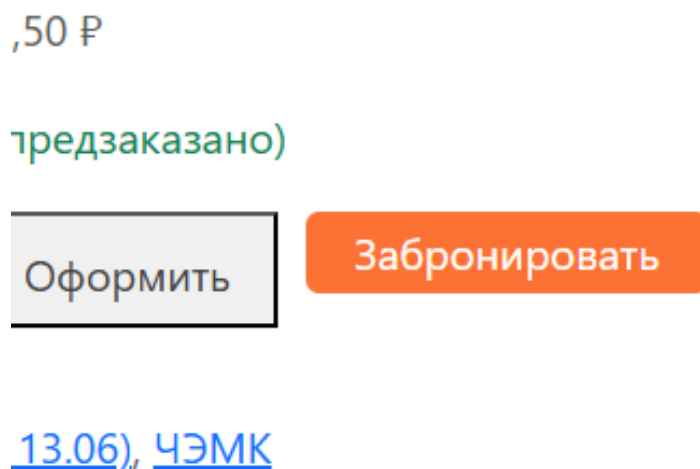


Рисунок 37 – Кнопка «забронировать».

Функция `render_booking_page()` - обрабатывает шорткод и отображает форму бронирования, получение товара по ID, обработка запроса бронирования, если форма отправлена. Подробнее на рисунке 38.

```
public function render_booking_page() {  
    if (!isset($_GET['product_id'])) {  
        return __('Товар не выбран.', 'woocommerce-booking');  
    }  
  
    $product_id = intval($_GET['product_id']);  
    $product = wc_get_product($product_id);  
  
    if (!$product) {  
        return __('Товар не найден.', 'woocommerce-booking');  
    }  
  
    $error_message = '';  
    if (isset($_POST['submit_booking'])) {  
        $error_message = $this->handle_booking_request($product_id);  
    }  
  
    return $this->get_booking_form($product, $error_message);  
}
```

Рисунок 38 – Обработка шорткода и отображение формы бронирования.

Функция `get_booking_form()` генерирует HTML-форму бронирования товара, а также после нажатия на кнопку «забронировать» появляется модальное окно с сообщением о просьбе перейти на электронную почту. Код представлен на рисунках 39 и 40.

```
private function get_booking_form($product, $error_message = '') {
    ob_start();
    ?>
    <div class="woocommerce-booking-form">
        <h2><?php echo __('Бронирование товара:', 'woocommerce-booking') . ' ' . $product->get_name(); ?></h2>

        <?php if ($error_message): ?>
            <div class="error-message" style="color:red;"><?php echo esc_html($error_message); ?></div>
        <?php endif; ?>

        <form method="post" id="booking-form" style="margin-bottom: 3rem;">
            <p class="form-row">
                <label for="billing_first_name"><?php _e('Ф.И.О. родителя или представителя полностью', 'woocommerce-booking'); ?>
                <abbr class="required" title="Required Field">*</abbr></label>
                <input type="text" id="billing_first_name" name="billing_first_name" required>
                <span class="error-message" id="billing_first_name_error" style="color:red;"></span>
            </p>
            <p class="form-row">
                <label for="billing_child_name"><?php _e('Ф.И.О. ребенка полностью', 'woocommerce-booking'); ?>
                <abbr class="required" title="Required Field">*</abbr></label>
                <input type="text" id="billing_child_name" name="billing_child_name" required>
                <span class="error-message" id="billing_child_name_error" style="color:red;"></span>
            </p>
            <p class="form-row">
                <label for="billing_wooccm11"><?php _e('Дата рождения ребенка', 'woocommerce-booking'); ?>
                <abbr class="required" title="Required Field">*</abbr></label>
                <input type="date" id="billing_wooccm11" name="billing_wooccm11" required>
                <span class="error-message" id="billing_wooccm11_error" style="color:red;"></span>
            </p>
            <p class="form-row">
                <label for="billing_email"><?php _e('Email', 'woocommerce-booking'); ?>
                <abbr class="required" title="Required Field">*</abbr></label>
                <input type="email" id="billing_email" name="billing_email" required>
                <span class="error-message" id="billing_email_error" style="color:red;"></span>
            </p>
            <p class="form-row">
                <label for="billing_phone"><?php _e('Телефон', 'woocommerce-booking'); ?><abbr class="required" title="Required Field">*</abbr></label>
                <input type="tel" id="billing_phone" name="billing_phone" required>
                <span class="error-message" id="billing_phone_error" style="color:red;"></span>
            </p>
            <button type="submit" name="submit_booking" class="button alt" style="background-color:#FC7234"><?php _e('Забронировать', 'woocommerce-booking'); ?></button>
        </form>
    </div>
}
```

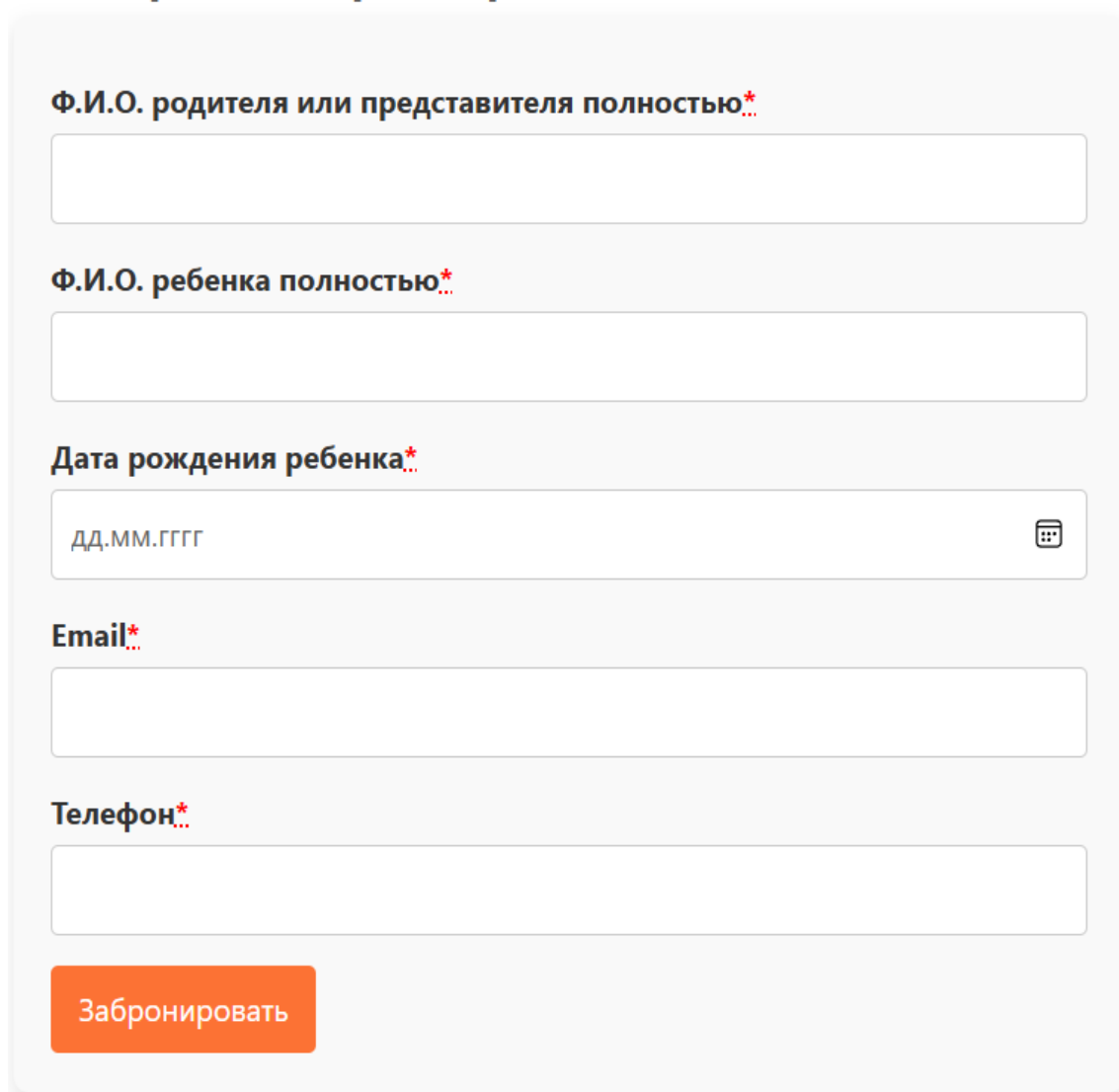
Рисунок 39 – Форма для бронирования.

```
<!-- Модальное окно -->
<div id="booking-success-modal" style="display:none;">
    <div class="modal-content">
        <span class="close-button">&times;</span>
        <p><?php _e('Бронирование успешно завершено! Пожалуйста, проверьте вашу почту для подтверждения. Если не увидели письма, то перейдите во вкладку "Спам".', 'woocommerce-booking'); ?></p>
    </div>
</div>
```

Рисунок 40 - Модальное окно.

					ДП.09.02.07.01.401.275.25.ПЗ	ЛИСТ
Изм.	Лист	№ докум.	Подпись	Дата		36

На рисунках 41 и 42 можно увидеть, как выглядит форма для бронирования путевки, а также модальное окно с важной информацией о последующих действиях.



The form is a light gray rounded rectangle containing several input fields and a button. At the top, it has a label 'Ф.И.О. родителя или представителя полностью*' followed by a text input field. Below this is another label 'Ф.И.О. ребенка полностью*' followed by a text input field. The third field is 'Дата рождения ребенка*' with a date picker icon on the right. The fourth field is 'Email*' followed by a text input field. The fifth field is 'Телефон*' followed by a text input field. At the bottom left of the form is an orange button with the text 'Забронировать'.

Рисунок 41 – Форма для бронирования на сайте.

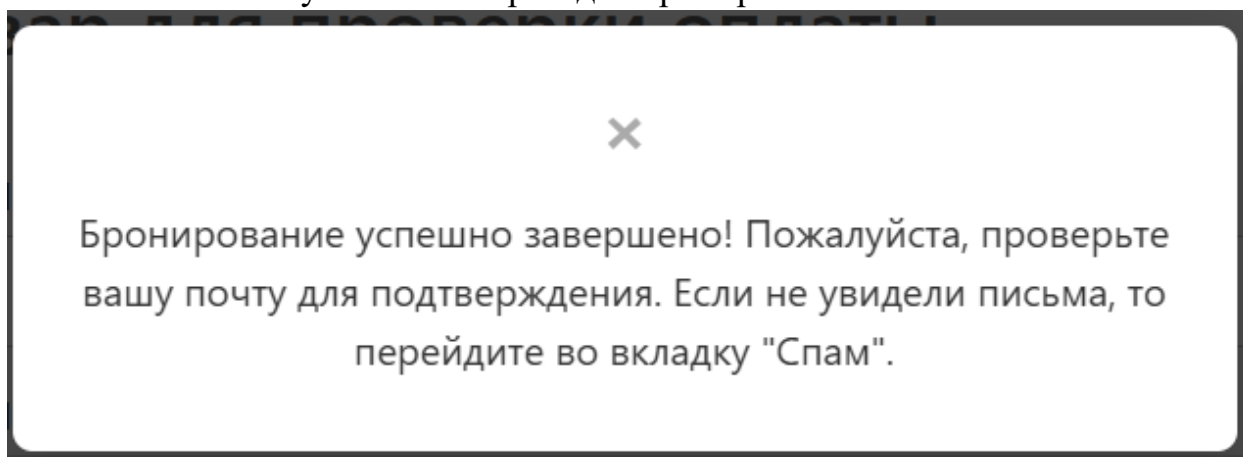


Рисунок 42 – Модальное окно на сайте.

Функция `generate_booking_token()` – генерирует уникальный токен подтверждения, код представлен на рисунке 43.

```
private function generate_booking_token() {  
    return bin2hex(random_bytes(16));  
}
```

Рисунок 43 – Уникальный код подтверждения.

Функция `handle_booking_request()` обрабатывает запрос на бронирование товара. Сначала проверяется доступность товара, валидируются обязательные поля, проверка на корректность данных, генерация токена подтверждения, уменьшение товара на складе, сохранение данных в БД, установка сроков подтверждения, возвращение ошибок, если есть и отправка email с ссылкой подтверждения. Код функционала на рисунках 44,45 и 46.

```
private function handle_booking_request($product_id) {  
    $product = wc_get_product($product_id);  
  
    if (!$product || !$product->is_in_stock()) {  
        return __('Товар недоступен для бронирования.', 'woocommerce-booking');  
    }  
  
    // Проверка обязательных полей  
    if (empty($_POST['billing_first_name']) || empty($_POST['billing_child_name']) ||  
        empty($_POST['billing_wooccm11']) || empty($_POST['billing_email']) ||  
        empty($_POST['billing_phone'])) {  
        return __('Все поля обязательны для заполнения.', 'woocommerce-booking');  
    }  
  
    $parent_name = sanitize_text_field($_POST['billing_first_name']);  
    $child_name = sanitize_text_field($_POST['billing_child_name']);  
    $child_bdate = sanitize_text_field($_POST['billing_wooccm11']);  
    $email = sanitize_email($_POST['billing_email']);  
    $phone = sanitize_text_field($_POST['billing_phone']);  
  
    // Валидация ФИО  
    if (!preg_match('/^[А-Я-яёЁ\s]+$/', $parent_name) || count(explode(' ', $parent_name)) < 3) {  
        return __('ФИО родителя должно содержать только кириллицу и быть в формате "Фамилия Имя Отчество".', 'woocommerce-booking');  
    }  
  
    if (!preg_match('/^[А-Я-яёЁ\s]+$/', $child_name) || count(explode(' ', $child_name)) < 3) {  
        return __('ФИО ребенка должно содержать только кириллицу и быть в формате "Фамилия Имя Отчество".', 'woocommerce-booking');  
    }  
  
    // Валидация телефона  
    if (!preg_match('/^\+7 \(\d{3}\) \d{3} \d{2} \d{2}$/', $phone)) {  
        return __('Неверный формат телефона.', 'woocommerce-booking');  
    }  
  
    // Валидация даты рождения  
    if (empty($child_bdate)) {  
        return __('Дата рождения ребенка обязательна.', 'woocommerce-booking');  
    }  
}
```

Рисунок 44 – Валидация полей и их проверка.

```

$confirmation_token = $this->generate_booking_token();

// Уменьшаем количество товара на складе
$current_stock = $product->get_stock_quantity();
if ($current_stock > 0) {
    $product->set_stock_quantity($current_stock - 1);
    $product->save();
} else {
    return __('Товар закончился на складе.', 'woocommerce-booking');
}

$booking_id = $this->log_booking($product_id, $product->get_name(),
    $parent_name, $child_name, $child_bdate, $email, $phone, $confirmation_token);

$subject = __('Подтверждение бронирования', 'woocommerce-booking');
$confirmation_url = home_url('/confirm-booking/?token=' . $confirmation_token);
$message = sprintf(
    __('Здравствуйте, %s!<br><br>Ваш товар "%s" успешно забронирован.<br><br>
    Пожалуйста, подтвердите бронирование в течение 24 часов, перейдя по ссылке: %s<br><br>
    После подтверждения вы получите ссылку для оплаты. Оплата должна быть произведена в течение 72 часов.', 'woocommerce-booking'),
    $parent_name,
    $product->get_name(),
    $confirmation_url
);

```

Рисунок 45 – Уменьшение товара на складе, установка срока подтверждения.

```

// Отправляем HTML-письмо
add_filter('wp_mail_content_type', function() { return 'text/html'; });
wp_mail($email, $subject, $message);
remove_filter('wp_mail_content_type', 'set_html_content_type');

return '';

```

Рисунок 46 – Отправка письма на email.

Как выглядит форма с некорректным вводом данных и как отображаются ошибки продемонстрировано на рисунке 47.

Ф.И.О. родителя или представителя полностью*

аваа

Ф.И.О. родителя должно содержать только кириллицу и быть в формате "Фамилия Имя Отчество".

Ф.И.О. ребенка полностью*

аа

Ф.И.О. ребенка должно содержать только кириллицу и быть в формате "Фамилия Имя Отчество".

Дата рождения ребенка*

28.01.2011



Email*

borovinskikhpolina@yandex.ru

Телефон*

+7 (777) 777 77 7

Неверный формат телефона.

Забронировать

Рисунок 47 – Некорректный ввод данных.

Функция `log_booking()` – сохраняет данные бронирования в базу данных. Код на рисунке 48.

					ДП.09.02.07.01.401.275.25.ПЗ	лист
Изм.	Лист	№ докум.	Подпись	Дата		40


```

private function log_booking($product_id, $product_name,
$parent_name, $child_name, $child_bdate, $email, $phone, $confirmation_token) {
    global $wpdb;
    $table_name = $wpdb->prefix . 'woocommerce_bookings';

    return $wpdb->insert(
        $table_name,
        array(
            'product_id'           => $product_id,
            'product_name'         => $product_name,
            'parent_name'          => $parent_name,
            'child_name'           => $child_name,
            'child_bdate'          => $child_bdate,
            'email'                 => $email,
            'phone'                 => $phone,
            'confirmation_token'    => $confirmation_token,
            'booking_date'          => current_time('mysql'),
        )
    );
}

```

Рисунок 48 – Функция log_booking()

Функция handle_booking_confirmation() обрабатывает подтверждение бронирования по токenu. В данной функции реализовано нахождение по токenu, проверка на доступность товара, создание заказа и обновление статуса, установка срока оплаты в течение 72 часов, отправка письма на email. Код функции на рисунках 49, 50, 51 соответственно.

```

public function handle_booking_confirmation() {
    //проверка на наличие токена
    if (isset($_GET['token'])) {
        global $wpdb;
        $table_name = $wpdb->prefix . 'woocommerce_bookings';
        $token = sanitize_text_field($_GET['token']);

        $booking = $wpdb->get_row($wpdb->prepare(
            "SELECT * FROM $table_name WHERE confirmation_token = %s",
            $token
        ));

        if ($booking) {
            // Проверяем наличие товара перед созданием заказа
            $product = wc_get_product($booking->product_id);
            if (!$product || !$product->is_in_stock()) {
                $subject = __('Бронирование отменено - товар недоступен', 'woocommerce-booking');
                $message = sprintf(
                    __('Здравствуйте, %s!<br><br>К сожалению, ваш забронированный товар "%s" больше недоступен, и ваше бронирование было отменено.', 'woocommerce-booking'),
                    $booking->parent_name,
                    $booking->product_name
                );

                add_filter('wp_mail_content_type', function() { return 'text/html'; });
                wp_mail($booking->email, $subject, $message);
                remove_filter('wp_mail_content_type', 'set_html_content_type');

                $wpdb->delete($table_name, array('id' => $booking->id));

                wp_die(
                    __('К сожалению, товар больше недоступен. Бронирование отменено.', 'woocommerce-booking'),
                    __('Ошибка', 'woocommerce-booking'),
                    array('response' => 404)
                );
            }
        }
    }
}

```

Рисунок 49 – Проверка на наличие токена и товара.

```

// Обновляем статус бронирования
$wpdb->update(
    $table_name,
    array(
        'is_confirmed' => 1,
        'payment_deadline' => time() + 72 * 3600
    ),
    array('id' => $booking->id)
);

// Создаем заказ WooCommerce
$order = wc_create_order();
$order->add_product($product, 1);

// Устанавливаем данные покупателя
$order->set_billing_email($booking->email);
$order->set_billing_first_name($booking->parent_name);
$order->set_billing_phone($booking->phone);

$order->calculate_totals();
$order->update_status('pending', __('Ожидает оплаты', 'woocommerce-booking'));

// Сохраняем ID заказа
$wpdb->update(
    $table_name,
    array('order_id' => $order->get_id()),
    array('id' => $booking->id)
);

```

Рисунок 50 – Обновление статуса бронирования и работа с заказом.

```

// Отправляем письмо со ссылкой на оплату
$payment_url = $order->get_checkout_payment_url();
$subject = __('Ваше бронирование подтверждено - ссылка для оплаты', 'woocommerce-booking');
$message = sprintf(
    __('Здравствуйте, %s!<br><br>Ваше бронирование товара "%s" подтверждено.<br><br>
    Для завершения бронирования, пожалуйста, оплатите заказ в течение 72 часов по ссылке: %s<br><br>
    Если вы не оплатите заказ в течение этого времени, бронирование будет автоматически отменено.', 'woocommerce-booking'),
    $booking->parent_name,
    $product->get_name(),
    $payment_url
);

add_filter('wp_mail_content_type', function() { return 'text/html'; });
wp_mail($booking->email, $subject, $message);
remove_filter('wp_mail_content_type', 'set_html_content_type');

wp_die(
    __('Бронирование успешно подтверждено! На ваш email отправлена ссылка для оплаты.', 'woocommerce-booking'),
    __('Подтверждение бронирования', 'woocommerce-booking'),
    array('response' => 200)
);
} else {
    wp_die(
        __('Недействительный токен подтверждения.', 'woocommerce-booking'),
        __('Ошибка', 'woocommerce-booking'),
        array('response' => 404)
    );
}

```

Рисунок 51 – Отправка письма со ссылкой на оплату

2.5 Алгоритм внедрения программного модуля в бизнес-процесс

Внедрение программного модуля онлайн-бронирования путевок для детского лагеря представляет собой многоэтапный процесс, требующий тщательного планирования, согласования с ключевыми заинтересованными сторонами и последующего контроля. Успешная интеграция данного решения позволит автоматизировать процесс бронирования, снизить нагрузку на административный персонал, минимизировать ошибки, связанные с ручным вводом данных, и повысить удобство для родителей, оформляющих путевки для своих детей.

Подготовительный этап:

На данном этапе осуществляется всесторонний анализ требований, оценка технической инфраструктуры, подготовка персонала и разработка необходимой документации.

Прежде всего, необходимо провести анализ функциональных и технических требований к модулю бронирования. Важно определить, какие именно функции должны быть реализованы. Также следует учесть

					ДП.09.02.07.01.401.275.25.ПЗ	лист
Изм.	Лист	№ докум.	Подпись	Дата		44

специфику работы детского лагеря – сезонность, ограниченное количество мест.

Параллельно проводится оценка инфраструктуры сайта. Поскольку модуль будет работать как WordPress-плагин, необходимо убедиться в совместимости с текущей версией CMS, а также проверить, не возникнет ли конфликтов с другими установленными плагинами (особенно если на сайте уже используются формы обратной связи, платежные системы или CRM). Отдельное внимание уделяется безопасности.

Одним из ключевых аспектов подготовительного этапа является обучение персонала. Администраторы сайта должны освоить управление модулем: просмотр отчета по заявкам(бронированиям), умение создать заявку от имени администратора, просмотр базы данных в настройках плагина. Для этого проводятся обучающие сессии, а также разрабатываются инструкции.

Завершающим шагом подготовительного этапа становится подготовка документации. Создается руководство для администратора, в котором подробно описывается процесс настройки модуля и устранения возможных неполадок. Для родителей разрабатывается пошаговая инструкция по бронированию путевки, оплате и получению подтверждающих документов.

Установка и настройка модуля:

После завершения подготовительных работ осуществляется развертывание модуля на сайте детского лагеря. Плагин устанавливается через административную панель WordPress, после чего производится его первоначальная настройка. Важно правильно настроить права доступа, чтобы только уполномоченные сотрудники могли вносить изменения в конфигурацию.

Далее выполняется детальное конфигурирование модуля. Подключаются email-уведомления – автоматическая отправка подтверждений брони, отправление ссылки для оплаты после подтверждения.

Перед запуском модуля в эксплуатацию проводится тщательное тестирование. Проверяется корректность работы всех функций: создание брони, видимость брони в панели у администратора, корректные переходы по ссылкам, экспорт данных о бронированиях в excel. Особое внимание уделяется нагрузочному тестированию, чтобы убедиться, что система справится с наплывом пользователей в период активного бронирования.

Запуск и сопровождение:

После успешного тестирования модуль переводится в эксплуатацию. На этом этапе доступ к онлайн-бронированию получает ограниченная группа

					ДП.09.02.07.01.401.275.25.ПЗ	лист
Изм.	Лист	№ докум.	Подпись	Дата		45

пользователей – отдел продаж, администратор. Это позволяет в реальных условиях проверить стабильность работы, выявить возможные недочеты и оперативно их исправить. Собирается обратная связь от пользователей, анализируются их замечания и пожелания.

Если пилотный запуск проходит успешно, модуль вводится в промышленную эксплуатацию. Теперь все родители могут бронировать путевки через сайт. Важно обеспечить техническую поддержку на первых порах – назначить ответственного за мониторинг работы системы, оперативное реагирование на возможные сбои и консультирование пользователей.

Для обеспечения долгосрочной стабильности работы модуля организуется регулярное сопровождение: обновление плагина, резервное копирование данных, мониторинг безопасности.

Оценка эффективности внедрения:

После нескольких недель или месяцев работы модуля проводится анализ его эффективности. Собирается обратная связь от родителей – насколько удобным оказался процесс бронирования, были ли технические сложности. Опрашиваются сотрудники лагеря – сократилось ли время обработки заявок, уменьшилось ли количество ошибок.

Сравниваются ключевые показатели до и после внедрения: количество оформленных онлайн-броней, среднее время обработки заявки, уровень удовлетворенности клиентов. На основе этих данных составляется отчет, в котором фиксируются достигнутые результаты и планируются дальнейшие улучшения системы.

2.6 Условия эксплуатации

Модуль онлайн-бронирования создан для работы с интернет-магазинами на WordPress и WooCommerce. Он позволяет клиентам бронировать товары прямо на сайте, а администраторам – управлять этими бронированиями.

Для работы модуля нужен сайт на WordPress с установленным WooCommerce. Сервер должен поддерживать PHP версии 7.4 или выше. Хорошо, если на сервере будет хотя бы 1 ГБ оперативной памяти и 10 ГБ места для хранения данных. Обязательно должен быть включен протокол HTTPS для безопасности данных.

После установки модуль сам создает нужную таблицу в базе данных. На страницах товаров появляется кнопка "Забронировать", которая ведет на форму бронирования. Клиент заполняет свои данные: ФИО родителя и

					ДП.09.02.07.01.401.275.25.ПЗ	лист
Изм.	Лист	№ докум.	Подпись	Дата		46

ребенка, дату рождения ребенка, email и телефон. Все поля проверяются на правильность заполнения. После отправки формы клиент получает письмо со ссылкой для подтверждения бронирования. Подтвердить бронь нужно в течение 24 часов, иначе она автоматически отменится. После подтверждения дается 72 часа на оплату.

Администратор видит все бронирования в специальном разделе админки. Там можно фильтровать записи, подтверждать или отменять брони, удалять их и экспортировать данные в Excel. Если бронь отменяется, товар возвращается на склад, а клиент получает уведомление на email.

Модуль автоматически проверяет бронирования каждый час. Неподтвержденные брони старше 24 часов и неоплаченные брони старше 72 часов удаляются, а товары возвращаются на склад. Клиенты получают об этом уведомления.

Для безопасности все личные данные клиентов защищены. Телефоны и email проверяются на правильность ввода. Письма отправляются в защищенном формате.

Модуль прост в использовании и не требует сложных настроек. Он помогает магазинам удобно организовывать бронирование товаров, а клиентам – легко бронировать нужные товары. Техническая поддержка гарантирует помощь в случае возникновения вопросов или проблем в течение года после установки.

					ДП.09.02.07.01.401.275.25.ПЗ	лист
Изм.	Лист	№ докум.	Подпись	Дата		47

3 РАСЧЕТ ЭКОНОМИЧЕСКОЙ ЭФФЕКТИВНОСТИ

3.1 Капитальные затраты

При разработке веб приложения экономическая эффективность определяется в результате расчетов капитальных и текущих (эксплуатационных) затрат до и после внедрения веб приложения.

Капитальные затраты представляют собой затраты на приобретение внеоборотных активов, а также на их модификацию (достройку, дооборудование, реконструкцию) и модернизацию.

Основной характеристикой капитальных затрат является продолжительность их использования.

Капитальные затраты рассчитываются по формуле 1.

$$K = K_{\text{зпл}} + K_{\text{а}} + K_{\text{э}}, \quad (1)$$

где:

K – капитальные затраты,

$K_{\text{зпл}}$ – затраты на заработную плату разработчику со страховыми взносами во внебюджетные фонды,

$K_{\text{а}}$ – затраты на амортизацию на период разработки,

$K_{\text{э}}$ – затраты на электроэнергию на период разработки.

3.1.1 Затраты на заработную плату разработчику

Заработная плата (оплата труда работника) – вознаграждение за труд в зависимости от квалификации работника, сложности, количества, качества и условий выполняемой работы, а также компенсационные выплаты (доплаты и надбавки компенсационного характера, в том числе за работу в условиях, отклоняющихся от нормальных, работу в особых климатических условиях и на территориях, подвергшихся радиоактивному загрязнению, и иные выплаты компенсационного характера) и стимулирующие выплаты (доплаты и надбавки стимулирующего характера, премии и иные поощрительные выплаты).

Страховые взносы – это платежи, которые государство взимает с работодателей на финансовое обеспечение реализации прав застрахованных лиц на получение обеспечения по тому или иному виду обязательного социального страхования.

Страховые взносы рассчитываются от фонда оплаты труда ФОТ в размере 30% с зарплаты и иных выплат сотрудникам, к которым относятся (ст. 420 Налогового кодекса):

– премии;

					ДП.09.02.07.01.401.275.25.ПЗ	лист
Изм.	Лист	№ докум.	Подпись	Дата		48

- отпускные и компенсации за неиспользованный отпуск;
- материальная помощь более 4000 руб. на одного сотрудника за год.

Формула расчёта заработной платы при повременно премиальной системе оплаты труда:

$$\text{Сзпл} = (\text{ЧТС} \times \text{Фэф}) \cdot (1 + \text{ппрем} \% / 100\%), \quad (2)$$

где:

Сзпл – расходы на заработную плату,

ЧТС – часовая тарифная ставка по трудовому договору,

Фэф – эффективный фонд отработанного времени,

ппрем – процент начисляемой премии.

Страховые взносы во внебюджетные формы считаются по формуле:

$$\text{Ссвз} = \text{Сзпл} \cdot 30\%, \text{ где}$$

Ссвз – взносы в страховые фонды,

Сзпл – расходы на заработную плату разработчику за период разработки.

Эффективный фонд: 15.12.2024 – 15.05.2025 (Итого 5 месяцев, 152 дня, 792 часов)

$$\text{Сзпл} = (284,09 \cdot 792) \cdot (1 + 40\% / 100\%) = 314998,99 \text{ (руб.)}$$

$$\text{Ссвз} = 314998,99 \cdot 30\% = 94499,7 \text{ (руб.)} - \text{страховые взносы.}$$

3.1.2 Затраты на амортизацию на период разработки

Амортизация – постепенный перенос стоимости основных средств производства на себестоимость продукции (по мере их материального износа или морального устаревания).

Суть амортизации в том, что организация благодаря амортизационным отчислениям постепенно возмещает стоимость объекта. Процесс может выглядеть следующим образом:

- Организация что-то приобретает профессиональное оборудование;
- Далее фиксируется полная стоимость покупки;
- Определяется наценка на товар или услугу, которые производит организация;
- Производятся регулярные отчисления.

Далее, когда актив изнашивается, его могут либо списать и заменить на новый, либо использовать дальше и получать уже чистую прибыль.

					ДП.09.02.07.01.401.275.25.ПЗ	лист
Изм.	Лист	№ докум.	Подпись	Дата		49

Процент отчислений на амортизацию зависит от типа износа. Если объект подвергается нескольким типам износа сразу (моральный и материальный), то коэффициент износа будет максимальным. Уровень износа напрямую зависит от интенсивности использования основных средств.

Сумма амортизации определяется по формуле 4:

$$СА = (Сп * На) / 100\%, \quad (3)$$

где:

СА – сумма амортизационных отчислений за год,

Сп – первоначальная стоимость основных фондов,

На – норма амортизации

Норма амортизации оборудования рассчитывается по формуле 4.

$$На = (1 / Та) * 100\%, \quad (4)$$

где:

На – норма амортизации (%),

Та – срок полезного использования оборудования в годах.

Норма амортизации и первоначальная стоимость представлены в таблице 1.

Таблица 1 – Норма амортизации и первоначальная стоимость оборудования.

№	Наименование оборудования	Первоначальная стоимость (руб.)	Норма амортизации
1	ПК РС-X4657900	60000	33
2	Монитор АОС	11000	33

Расчет нормы амортизации:

ПК: $(1/3)*100\%=33\%$

Монитор: $(1/3)*100\%=33\%$

Расчет суммы амортизации на период разработки 5 месяцев:

ПК = $((60000*33)/100\%)*5/12= 8250$ (руб.)

Монитор = $((11000*33)/100\%)*5/12= 1512,5$ (руб.)

СП= $8250+1512,5 = 9\,762,5$ (руб.)

Итого сумма амортизационных отчислений за 5 месяцев составила:

9 762,5 (руб.)

3.1.3 Затраты на оборудование

Перечень оборудования представлен в таблице 1.

Затраты на оборудование не включаются в расчет капитальных вложений, так как оно было приобретено до начала разработки.

3.1.4 Затраты на электроэнергию

Затраты на электроэнергию при работе оборудования для технологических целей рассчитывают по формуле 5.

$$\text{Эоб} = \text{Роб} * \text{Цэ} * \text{тоб}, \quad (5)$$

где:

Эоб – затраты на электроэнергию, потребляемую оборудованием, руб.,

Роб – мощность, потребляемая всем оборудованием, кВт,

Цэ – тарифная ставка за 1кВт/час, Цэ = 5,62 руб.,

тоб – время работы оборудования, час.

Мощность, потребляемая оборудованием для разработки системы на период разработки представлена в таблице 2.

Таблица 2 – Мощность, потребляемая оборудованием для разработки системы на период разработки.

Название прибора	Электропотребление, Вт	Энергопотребление, кВт	Время работы, ч/сутки
ПК РС-Х4657900	450	0,45	8
Монитор АОС	20	0,2	8

Расчет затрат на электроэнергию на период разработки 4 месяца:

ПК: $0,45 * 8 * 5,62 * 152 = 3075,26$ (руб.)

Монитор: $0,2 * 8 * 5,62 * 152 = 1\,366,78$ (руб.)

$\text{Эоб} = 3075,26 + 1366,78 = 4\,442,04$ (руб.)

Итого затраты на электроэнергию на разработку: Эоб = 4 442,04 (руб.)

Итого капитальные затраты на обновление системы составили:

$K = 314998,99 + 94499,7 + 9\,762,5 + 4\,442,04 = 423\,703,23$ (руб.)

3.2 Эксплуатационные затраты (текущие затраты)

Эксплуатационные затраты (текущие затраты) – абсолютный показатель издержек, понесенных предприятием на обеспечение работоспособности производственных фондов.

Эксплуатационные затраты считаются по формуле 6.

$$C = C_c + C_a + C_{зпл} + C_{свз} + C_{эл} + C_n + C_{нма} + C_{п}, \quad (6)$$

где:

C_c – затраты на сырье, материалы, комплектующие,

C_a – затраты на амортизацию за год,

$C_{зпл}$ – затраты на заработную плату сотрудников за год,

$C_{свз}$ – страховые взносы во внебюджетные фонды,

$C_{эл}$ – затраты на электроэнергию за год,

C_n – накладные затраты (среднеотраслевые расходы составляют от 80 до 120 % от годового фонда оплаты труда сотрудников),

$C_{нма}$ – затраты на нематериальные активы,

$C_{п}$ – затраты на простои оборудования.

3.2.1. Затраты на сырье, материалы и комплектующие.

Затраты на материалы определяются общей суммой затрат на все материалы, запасные части и т.д. (данные договора купли-продажи, сметы и т.д.).

Однако, модуль онлайн-бронирования – это новшество для сайта детского оздоровительного лагеря АО «ЧЭМК», поэтому капитальные затраты отсутствовали.

Итого затраты на материалы: 0 (руб.)

После внедрения:

После внедрения модуля какие-либо материалы не требуются.

Итого затраты на материалы: 0 (руб.)

3.2.2 Страховые взносы во внебюджетные формы

Затраты на оплату труда сотрудников после внедрения представлены в таблице 3.

Таблица 3 – Затраты на оплату труда сотрудников за год, после внедрения.

Должность	Численность	Оклад, руб./час	Премия, %	Затраты на оплату труда за год, руб.
Администратор сайта	1	243,9	40	672676
ИТОГО годовой ФОТ:	672676			

Расчет годового ФОТ:

Администратор сайта: $((243,9 \cdot 1970) \cdot 1,4) \cdot 1 = 672\,676$ (руб.)

Итого ФОТ: 672676 (руб.)

Расчет страховых взносов после внедрения:

Ссвз = $672676 \cdot 30\% = 201\,802,8$ (руб.) – страховые взносы во внебюджетные формы сотрудников за год.

Затраты на электроэнергию за год.

Мощность, потребляемая оборудованием после внедрения представлена в таблице 4.

Таблица 4 – Мощность, потребляемая оборудованием после внедрения.

Название прибора	Электропотребление, Вт	Энергопотребление, кВт	Время работы, ч/сутки
ПК РС-Х4657900	450	0,45	8
Монитор АОС	20	0,2	8

Затраты на энергопотребление системы после внедрения:

ПК = $0,45 \cdot 8 \cdot 5,62 \cdot 365 = 7\,384,68$ (руб.)

Монитор = $0,2 \cdot 8 \cdot 5,62 \cdot 365 = 3\,282,08$ (руб.)

Итого затраты на электроэнергию после внедрения: 10 666,76 (руб.)

Накладные затраты

Накладные расходы – расходы, которые непосредственно не связаны с производственной деятельностью компании, однако обеспечивают непрерывность и качество производственного процесса. Накладные расходы являются дополнительными расходами компании. Они не связаны с производством напрямую, т. е. не включаются в себестоимость выпускаемой продукции или оказываемых услуг. Их невозможно отнести на конкретный

вид продукции, работы или услуг. Это сопутствующие траты, которые обеспечивают производственный процесс.

Какие затраты стандартно включаются в накладные расходы:

- амортизация основных средств, которые не используются в производственном процессе;
- на приобретение оборудования, инструментов для производства;
- на содержание и аренду площадей, неиспользуемых в производстве, офисных и торговых помещений;
- на содержание административно-управленческого и общепроизводственного персонала;
- на рекламу;
- приобретение расходных материалов для офиса;
- телекоммуникационные расходы;
- транспортные расходы;
- выплаты в ФСС, ПФР;
- оплату сопутствующих услуг (консультационных, информационных, юридических);
- командировочные и представительские расходы.

Накладные расходы – это издержки компании, которые не относятся напрямую к технологическому процессу изготовления товара, но обеспечивают его, хотя не входят в стоимость сырья и работы.

Пропорция между прямыми затратами и накладными расходами может устанавливаться от разных показателей прямых затрат.

Накладные расходы считаются по формуле 7.

$$C_n = C_{зпл} * \%, \quad (7)$$

где:

$C_{зпл}$ – расходы на заработную плату сотрудникам за год, % – процент накладных расходов.

Процент накладных расходов равен 100.

Итого: $C_n = 672676 * 100\% = 6\,726,76$ (руб.)

Затраты на нематериальные активы.

Нематериальные активы (НМА) – это объекты интеллектуальной собственности, которые используются более 1 года и приносят доход. При этом они не имеют материальной формы и отделимы от других активов.

Чтобы признать нематериальный актив, должны соблюдаться все условия из положения по бухгалтерскому учету ПБУ 14/2007:

					ДП.09.02.07.01.401.275.25.ПЗ	лист
Изм.	Лист	№ докум.	Подпись	Дата		54

– Экономическая выгода. Бизнес использует нематериальный актив для производства продукции, торговли товарами, выполнения работ, оказания услуг или управления.

– Длительное пользование. Нематериальный актив должен служить минимум 12 месяцев.

– Возможность отделить актив от других активов. Его можно использовать сам по себе, например продать.

– Нет материальной формы. Нематериальный актив – результат интеллектуальной деятельности.

Что относится к НМА

– компьютерные программы;

– изобретения и промышленные модели;

– научные, литературные произведения, произведения искусства;

– товарные знаки;

– ноу-хау (производственные секреты);

– деловая репутация.

Что не относится к НМА

– финансовые вложения;

– организационные расходы, связанные с созданием юридического лица;

– квалификация сотрудников, их профессиональные и деловые качества;

– научно-исследовательские и опытно-конструкторские работы (НИОКР) не приведшие к положительным результатам, не законченные, неоформленные НИОКР;

– носители информации, на которые записаны объекты интеллектуальной собственности.

НМА учитываются по первоначальной стоимости, амортизируются и отражаются на текущую дату с учётом морального износа.

До внедрения:

До внедрения покупка дополнительных нематериальных активов не требовалась.

СНМА=0 руб.

Таблица 5 – Расчет стоимости и амортизационных отчислений нематериальных активов после внедрения.

Наименование	Первоначальная стоимость, руб.	Балансовая стоимость, руб.	Норма амортизации, %	Годовая сумма амортизации, руб.
Windows 10 Pro	8 900	5 340	10	890
Visual Studio Code	0	0	0	0
ИТОГО:	890			

$СНМА=8900\text{руб.}$

Расчет стоимости и амортизационных отчислений нематериальных активов после внедрения.

После внедрения покупка дополнительных нематериальных активов не требуется.

Итого эксплуатационные затраты до внедрения = 0 (руб.)

Итого эксплуатационные затраты после внедрения:

$0+672\,676+10\,666,76+8900=692\,242,76\text{ (руб.)}$

Расчет экономического эффекта:

Экономический эффект – это конечный экономический результат, полученный от проведения какого-либо мероприятия, вызывающего улучшение каких-либо показателей работы организации. Результат является абсолютным показателем и измеряется в денежных или натуральных единицах.

В целом, получение эффекта предполагает первоначальное осуществление каких-либо затрат, а затем получение дополнительного дохода от внедрения мероприятия.

Сам экономический эффект представляет собой этот дополнительный доход, получаемый через дополнительные прибыли, снижение материальных, трудовых затрат, рост объемов производства или качества продукции, выражаемых в цене.

Экономическая эффективность – это относительный показатель доходов и расходов целого производства или определенного проекта. Это достижение желаемых результатов с минимальными усилиями, затратами или отходами.

Прирост прибыли за счет увеличения продаж путевок рассчитывается по формуле 10.

Продажи путевок до внедрения составили: 1500 (шт.)

Средняя цена путёвок: 35 000 (руб.)

Выручка рассчитывается по формуле 8:

$$V = P * Ц, \quad (8)$$

где:

V- выручка,

P- кол-во проданных путевок,

Ц- цена путевки.

$$V = 1500 * 35000 = 52\,500\,000 \text{ (руб.)}$$

Итого выручка от продажи путевок до внедрения: 52 500 000 (руб.)

После внедрения:

Продажи путевок после внедрения составили: 1570 (шт.)

Средняя цена путёвок: 35 000 (руб.)

$$V = 1570 * 35000 = 54\,950\,000 \text{ (руб.)}$$

Итого выручка от продажи путевок до внедрения: 54 950 000 (руб.)

Прирост выручки рассчитывается по формуле 9:

$$\Delta V = V_{\text{после}} - V_{\text{до}}, \quad (9)$$

где:

V_{после}- выручка после внедрения,

V_{до}- выручка до внедрения.

$$\Delta V = V_{\text{после}} - V_{\text{до}} = 54\,950\,000 - 52\,500\,000 = 2\,450\,000 \text{ (руб.)}$$

Итого прирост выручки после внедрения: 2 450 000 (руб.)

Экономический эффект – это чистая дополнительная прибыль от внедрения системы. Рассчитывается по формуле 10:

$$\Xi = \Delta V - C, \quad (10)$$

где:

ΔV - прирост выручки,

C - годовые эксплуатационные затраты

$$\Xi = 2\,450\,000 - 692\,243 = 1\,757\,757 \text{ (руб.)}$$

Итого годовой экономический эффект о внедрения = 1 757 757 (руб.)

Коэффициент экономической эффективности определяется по следующей формуле 11.

$$E = \Xi / K, \quad (11)$$

где:

E – экономическая эффективность,

Ξ – экономический эффект (оптимизация или прирост прибыли),

K – капитальные затраты на проект.

					ДП.09.02.07.01.401.275.25.ПЗ	лист
Изм.	Лист	№ докум.	Подпись	Дата		57

$$E = 1\,757\,757 / 423\,703,23 = 4,2$$

Итого коэффициент экономической эффективности= 4,2

Срок окупаемости представляет собой отрезок времени, после которого доход от проекта становится равен сумме вложенных денег.

То есть коэффициент срока окупаемости при инвестировании в какое-либо дело будет показывать, сколько нужно будет времени, чтобы вернуть вложенный капитал.

Срок окупаемости рассчитывается по формуле 12.

$$T_{ок} = K / Э, \quad (12)$$

где:

$T_{ок}$ – срок окупаемости, выраженный в годах,

K – сумма вложенных средств, капитальные затраты,

$Э$ – экономический эффект.

$$T_{ок} = 423\,703,23 / 1\,757\,757 = 0,24 \text{ (2,9 месяца или 88 дней).}$$

4 ОРГАНИЗАЦИЯ РАБОЧЕГО МЕСТА И ОХРАНА ТРУДА

Организация рабочего пространства и соблюдение норм охраны труда являются ключевыми факторами обеспечения безопасности, комфорта и продуктивности сотрудников. В АО «Челябинский Электрометаллургический Комбинат» особое внимание уделяется эргономике, правильному освещению, поддержанию порядка на рабочем месте, а также снижению физической и психологической нагрузки.

Охрана труда на предприятии направлена на предотвращение травматизма и профзаболеваний за счет соблюдения нормативов безопасности, регулярного обучения персонала и использования средств индивидуальной защиты. Грамотная организация труда и системный контроль за охраной труда способствуют не только сохранению здоровья работников, но и созданию эффективной рабочей среды, что положительно влияет на результаты компании.

Обязанности работодателя включают обеспечение безопасных условий труда, проведение обучения по охране труда, а также организацию специальной оценки условий труда (СОУТ) без ограничения по сроку действия. Кроме того, работодатель обязан контролировать состояние рабочих мест, проводить профилактику аварийных ситуаций, а также расследовать и учитывать несчастные случаи на производстве.

Обязанности работника, в том числе веб-разработчика, предусматривают обязательное прохождение вводного инструктажа по охране труда, обучение правилам безопасности на рабочем месте и противопожарный инструктаж. Сотрудник обязан соблюдать нормы охраны труда, правильно эксплуатировать оборудование, своевременно оповещать о неисправностях, а также соблюдать режим труда и отдыха, включая 15-минутные перерывы каждый час.

Основные опасные факторы для веб-разработчика связаны с работой за компьютером. К ним относятся воздействие электромагнитного излучения, повышенная нагрузка на зрение и статическое напряжение мышц. В соответствии с требованиями электробезопасности, максимально допустимый ток составляет 0,3 мА, а непрерывная работа за компьютером не должна превышать 6 часов без перерывов.

Режим работы за компьютером регламентируется санитарными нормами. Сотрудник должен делать перерывы каждые 1-2 часа продолжительностью 10-15 минут, а общее время работы за компьютером не должно превышать 7-8 часов в день. Для снижения нагрузки на зрение

					ДП.09.02.07.01.401.275.25.ПЗ	лист
Изм.	Лист	№ докум.	Подпись	Дата		59

необходимо использовать мониторы с антибликовым покрытием, а также соблюдать расстояние от глаз до экрана не менее 50-70 см.

Освещение рабочего места должно быть равномерным, без резких бликов и контрастов. Рекомендуется использовать естественное освещение в сочетании с искусственным, при этом свет должен падать сбоку, чтобы избежать отражений на экране. Уровень освещенности рабочей поверхности должен составлять не менее 300-500 люкс.

Ответственность за нарушения норм охраны труда может быть как административной, так и дисциплинарной. Работодатель может быть привлечен к ответственности в виде штрафа до 80 000 рублей для организации и до 10 000 рублей для должностных лиц. Работник, в свою очередь, несет дисциплинарную ответственность за несоблюдение требований охраны труда.

Соблюдение этих норм в АО «Челябинский Электрометаллургический Комбинат» способствует безопасной и эффективной работе всех сотрудников, включая веб-разработчиков, обеспечивая их здоровье и высокую производительность труда.

					ДП.09.02.07.01.401.275.25.ПЗ	лист
Изм.	Лист	№ докум.	Подпись	Дата		60

ЗАКЛЮЧЕНИЕ

В рамках дипломного проекта был разработан модуль онлайн-бронирования путевок для детского оздоровительного лагеря АО «Челябинский Электрометаллургический Комбинат». Внедрение данной системы позволило автоматизировать процесс оформления заявок, упростить взаимодействие с родителями и администрацией лагеря, а также повысить прозрачность и скорость обработки бронирований.

Особое внимание было уделено удобству интерфейса, безопасности данных и соответствию современным цифровым стандартам. Разработанный модуль включает интуитивную систему фильтрации, онлайн-бронирование и автоматическое подтверждение бронирования, что значительно сокращает временные затраты и минимизирует ошибки при ручном вводе информации.

Кроме того, в проекте учтены требования к организации рабочего места разработчика и соблюдению норм охраны труда, что обеспечивает безопасные и комфортные условия труда при дальнейшей поддержке и развитии платформы.

Реализация данного решения способствует повышению эффективности работы лагеря, снижению административной нагрузки и улучшению качества обслуживания клиентов. В перспективе рекомендуется расширять функционал системы, внедряя дополнительные сервисы (например, личный кабинет для родителей, интеграцию с CRM, аналитику загрузки лагеря), а также проводить регулярный мониторинг удобства пользователей и адаптировать платформу под меняющиеся потребности.

Проведённая работа подтверждает важность цифровизации процессов в сфере детского отдыха и демонстрирует, что автоматизация бронирования путёвок не только оптимизирует внутренние процессы, но и повышает привлекательность лагеря для родителей и детей. Данный проект может стать основой для дальнейшего развития цифровой инфраструктуры АО «ЧЭМК» и внедрения инновационных решений в корпоративной социальной политике компании.

					ДП.09.02.07.01.401.275.25.ПЗ	лист
Изм.	Лист	№ докум.	Подпись	Дата		61

СПИСОК ИСПОЛЬЗОВАННЫХ ИСТОЧНИКОВ

1. Зверева В.П. Назаров А.В. Сопровождение и обслуживание программного обеспечения компьютерных систем. – Издательство Академия, 2020. – 256 с.
2. Илюшечкин, В. М. Основы использования и проектирования баз данных: учебник для среднего профессионального образования / В. М. Илюшечкин. – испр. и доп. – Москва: Издательство Юрайт, 2024. – 213 с. – Текст: электронный // Образовательная платформа Юрайт [сайт]. – URL: <https://urait.ru/bcode/538545> (дата обращения: 26.03.2024).
3. Нестеров, С. А. Базы данных: учебник и практикум для среднего профессионального образования / С. А. Нестеров. – 2-е изд. – Москва: Издательство Юрайт, 2024. – 258 с. – Текст: электронный // Образовательная платформа Юрайт [сайт]. – URL: <https://urait.ru/bcode/542800> (дата обращения: 26.03.2024).
4. Полуэктова, Н. Р. Разработка веб-приложений: учебное пособие для среднего профессионального образования / Н. Р. Полуэктова. – 2-е изд. – Москва: Издательство Юрайт, 2024. – 204 с. – Текст: электронный // Образовательная платформа Юрайт [сайт]. – URL: <https://urait.ru/bcode/545237> (дата обращения: 28.03.2024).
5. Сысолетин, Е. Г. Разработка интернет-приложений: учебное пособие для среднего профессионального образования / Е. Г. Сысолетин, С. Д. Ростунцев. – Москва: Издательство Юрайт, 2023. – 90 с. – Текст: электронный // Образовательная платформа Юрайт [сайт]. – URL: <https://urait.ru/bcode/517538> (дата обращения: 28.03.2024).
6. Зараменских, Е. П. Информационные системы в бизнесе: учебник и практикум для среднего профессионального образования / Е. П. Зараменских. – 2-е изд., перераб. и доп. – Москва: Издательство Юрайт, 2024. – 470 с. —Текст: электронный // Образовательная платформа Юрайт [сайт]. – URL: <https://urait.ru/bcode/542802> (дата обращения: 26.03.2024). Федорова, Г.Н. Информационные системы: Учебник / Г.Н. Федорова. - М.: Academia, 2018. - 384 с.
7. Онлайн учебник по языку PHP – URL: <https://code.mu/ru/php/book/prime/>— Текст: электронный
8. Онлайн-учебник Laravel – URL: <https://code.mu/ru/php/framework/laravel/book/prime/>— Текст: электронный
9. Онлайн-учебник JavaScript – URL: <https://code.mu/ru/javascript/book/prime/>— Текст: электронный
10. Тузовский, А. Ф. Проектирование и разработка web-приложений: учебное пособие для среднего профессионального образования / А. Ф. Тузовский. – Москва: Издательство Юрайт, 2024. – 219 с. – Текст: электронный // Образовательная платформа Юрайт [сайт]. – URL: <https://urait.ru/bcode/541917> (дата обращения: 28.03.2024).

ПРИЛОЖЕНИЕ А. ТЕХНИЧЕСКОЕ ЗАДАНИЕ

					ДП.09.02.07.01.401.275.25.ПЗ	ЛИСТ
						63
Изм.	Лист	№ докум.	Подпись	Дата		

ОБЩЕЕ СОБРАНИЕ АКЦИОНЕРОВ (ЕДИНСТВЕННЫЙ АКЦИОНЕР)

