

## СОДЕРЖАНИЕ

ВВЕДЕНИЕ .....	4
1 ОБЩАЯ ЧАСТЬ .....	6
1.1 Цель разработки.....	6
1.2 Анализ технологий и возможных средств решения проблемы ..	7
1.3 Выбор средств и технологий .....	8
2 СПЕЦИАЛЬНАЯ ЧАСТЬ.....	11
2.1 Моделирование диаграммы потоков данных .....	14
2.2 Проектирование базы данных .....	15
2.3 Проектирование программного обеспечения .....	16
2.4 Тестирование и отладка .....	16
3 ЭКОНОМИЧЕСКАЯ ЧАСТЬ .....	21
3.1 Расчет себестоимости.....	21
3.2 Определение трудоемкости .....	22
3.2.1 Расчет материальных затрат .....	23
3.2.2 Расчет затрат на электроэнергию амортизации оборудования	24
3.2.3 Расчет расходов на заработную плату .....	24
3.2.4 Расчет себестоимости и цены разработки .....	25
3.2.5 Расчет экономического эффекта .....	25
4 ОБЕСПЕЧЕНИЕ БЕЗОПАСНОСТИ.....	25
4.1 Разработка мер защиты информации от несанкционированного доступа.....	26
4.2 Мероприятия по обеспечению безопасности жизнедеятельности .....	26
ЗАКЛЮЧЕНИЕ.....	27

					09.02.03.ИТ.ПДП.О			
Изм.	Лист	№ докум.	Подпись	Дат				
Разраб.	Скореев				КОЛЛЕДЖ НТИ НИЯУ МИФИ КПР-47Д			
Провер.	Горницкая							
Реценз								
Н. Контр.								
Утверд.								
					Лит.	Лист	Листов	
						2	59	

СПИСОК ИСПОЛЬЗОВАННЫХ ИСТОЧНИКОВ .....30

ПРИЛОЖЕНИЕ А.....30

ПРИЛОЖЕНИЕ В.....31

ПРИЛОЖЕНИЕ С.....31

## ВВЕДЕНИЕ

За последние пару лет в связи с развитием цифровой экономики и технологии, любому предприятию без представления себя в интернете не обойтись.

Основой разработки веб-сервиса является проблема отсутствия модуля создания и оплаты заказа на предприятии «Фото-Новик», являющегося партнером «Папа-Принтер».

До внедрения разработки возникала проблема в виде реальных неудобств и жалоб клиентов, которые приходили на предприятие и хотели сразу получить желаемый товар, однако прежде чем его получить необходимо было создать макет для печати и получить заказ только в следующий раз в назначенную дату. Безусловно создание макета и реализация товара отнимала много времени, а клиентопоток только увеличивался, что в результате сказывалось на качестве продукции и качества обслуживания.

Благодаря данной разработки предприятие сможет повысить качество производимой продукции за счет большего внимания к ней, что позволит достичь высоких технико-экономических показателей, экономить ресурсы, и более полно удовлетворить общественные потребности [1].

Таким образом, после внедрения результатов выпускной квалификационной работы на предприятие, клиенты смогут создавать макет для печати в удобном для них месте, оплачивать и приходить лично, отправив знакомого или получить товар от доставщика в назначенную администратором предприятия дату.

Ранее для увеличения конверсии на предприятиях «Папа-Принтер» и «Фото-Новик» был создан веб-ресурс для привлечения клиентов, а в настоящий момент в него будет внедрен веб-сервис для создания и оплаты заказов.

					09.02.03.ИТ.ПДП.О	Лист
						4
Изм.	Лист	№ докум.	Подпись	Дата		

Безоговорочно данный веб-сервис увеличит количество клиентов, и будет мотивировать заказывать куда больше, чем при личной встрече на предприятии, ведь когда человек в комфортной обстановке, он волен придумывать потрясающие глаз и воображение шедевры.

Подводя итоги, цель выпускной квалификационной работы является разработка «Проектирование и разработка веб-сервиса приема и оплаты заказов на примере сервисного центра «Папа-Принтер».

					09.02.03.ИТ.ПДП.О	Лист
						5
Изм.	Лист	№ докум.	Подпись	Дата		

# 1 ОБЩАЯ ЧАСТЬ

## 1.1 Цель разработки

Сформулируем цель разработки: «Распределить нагрузку на предприятии за счет веб-сервиса создания и оплаты заказов онлайн без участия сотрудников».

Предприятие «Фото-Новик» является партнером «Папа-Принтер» и занимается печатью на различных предметах, например, кружка или футболка.

Для выявления недостатков или преимуществ, произведем сравнение с веб-сервисами, которые находятся на первой позиции в поисковых системах. Одним из таких веб-сервисов является «vsemauykiru», предоставляющий печать практически на всём, что только можно себе представить, возможность создавать свой собственный дизайн, создавать заказ онлайн и ждать доставки

Наверняка в развитии сервиса «vsemauykiru» помог рост качества выпускаемой продукции, удобство создания заказа клиента и быстрая доставка, чего не достаёт предприятию «Фото-Новик».

Таким образом, в проекте должны быть реализованы функции оформления заказа, которые заключаются в создании макета для печати и заполнении контактных данных, а также функции оплаты и отправки чека на почту. Подробнее список функций отображен в списке ниже:

- Добавление текста на макет
- Добавление изображений на макет
- Перемещение текста и изображений
- Изменение размера текста и изображений
- Изменение цвета текста

					09.02.03.ИТ.ПДП.О	Лист
						6
Изм.	Лист	№ докум.	Подпись	Дата		

- Ввод email, телефона и имени для оформления заказа
- Оплата используя карту или qiwi кошелек
- Отправка письма с заказом на почту администратору
- Отправка письма с чеком на почту клиенту

Описанные выше цели были выявлены в процессе разработки, поскольку она велась с использованием гибких подходов.

Выбор методологии разработки основывался на размере предприятия и том что классические методологии подходят для более крупных проектов и не являются оптимальным решением поставленных задач. Выбор методологии будет подробнее описан в пункте 1.2.

Поскольку предприятие «Фото-Новик» не большое, то принято решение разрабатывать проект основываясь на гибких методологиях разработки, таких как Scrum [3] и Kanban [4], суть которых можно описать цитатой Клода Адриана Гельвеция: «Знание некоторых принципов легко возмещает незнание некоторых фактов» [5].

Благодаря использованию данных методологий разработки, удастся достичь необходимой гибкости в разработке программного продукта. Таким образом, проект будет готов практически к любым желаемым изменениям, и что более важно, намного быстрее получить функциональный продукт.

## 1.2 Анализ технологий и возможных средств решения проблемы

Исходя из сформулированных цели разработки, можно сделать вывод, что наиболее подходящей методологии разработки является гибкая с использованием двух подходов: Scrum и Kanban.

Поверхностно опишем для чего нужны данные подходы:

- Scrum – для разделения разработки на рабочие части

					09.02.03.ИТ.ПДП.О	Лист
						7
Изм.	Лист	№ докум.	Подпись	Дата		

– Kanban – для визуализации рабочего процесса

Таким образом, разработка будет разбита на независимые части.

Использование иных методологий является заведомо проигрышным решением, поскольку в условиях ограниченного времени и непостоянности в желаниях заказчика, они могут сгубить проект.

Например, при использовании каскадной модели разработки, проект может получиться нерасширяемым как монолит. Спиральная модель отлично подходит для дорогих и крупных проектов, где нет права на ошибку, так как на каждом этапе оцениваются риски [6].

Гибкая методология собрала в себя преимущества всех ранее существующих и позволяет использовать различные подходы одновременно, даже изменяя различные этапы, благодаря чему может подойти практически под любой небольшой и средний проект.

### 1.3 Выбор средств и технологий

Разрабатываемый веб-сервис будет внедрен в уже существующий веб-ресурс, который написан с использованием следующих технологий:

- Gatsby
- GraphQL
- ContentfulCMS
- Netlify

Веб-ресурс разрабатывался, используя подход JAMstack [7]. Если описывать коротко, то данный подход подразумевает разработку веб-приложения на желаемом фреймворке, управлять контентом из CMS и в результате получать HTML страницы, которые были сгенерированы с использованием рекомендаций SEO, PWA и Accessibility (A11y).

					09.02.03.ИТ.ПДП.О	Лист
						8
Изм.	Лист	№ докум.	Подпись	Дата		

Александр Товмач в своем выступлении на JS Fest 2019 года предоставил сравнение трех подходов в веб-разработке, которые будут кратко представлены ниже [16].

Ранее использовался PHP сервер-рендеринг, позволяющий извлекать данные из базы данных и, используя конструкции языка, вставлять их HTML шаблоны, которые в последствии предоставлялись пользователю в браузере. Схематично, это отображено на рисунке 1 ниже.

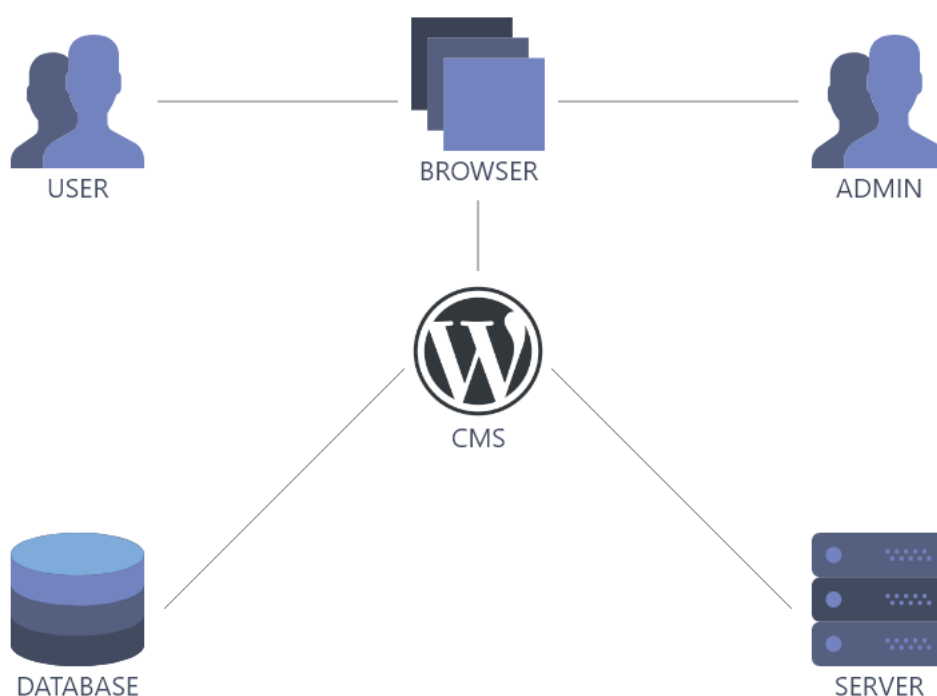


Рисунок 1 – PHP сервер-рендеринг

Впоследствии появилась JavaScript библиотека React, а также фреймворки Vue и Angular. Они предоставляют возможность разрабатывать быстрые динамичные интерфейсы, логика которых исполнялась на клиенте, а доступ к серверу осуществлялся через различные API. Подход отображен в виде схемы на рисунке 2.



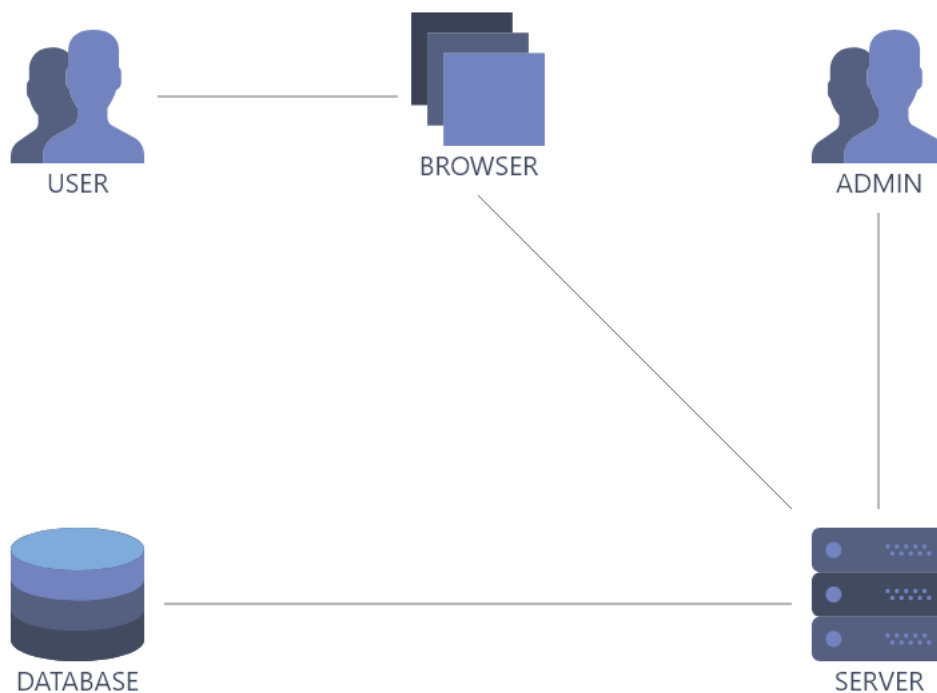


Рисунок 2 – Подход с использованием React, Vue или Angular

На рисунке 3 представлен JAM подход, который предполагает отказ от рендеринга на стороне сервера и клиента, компилировать весь сайт в готовые HTML страницы единожды при изменении контента, а также размещать всё на абсолютно любом файловом хостинге. Таким образом, при грамотной разработке, производительность измеряется только в скорости интернет-соединения пользователя.

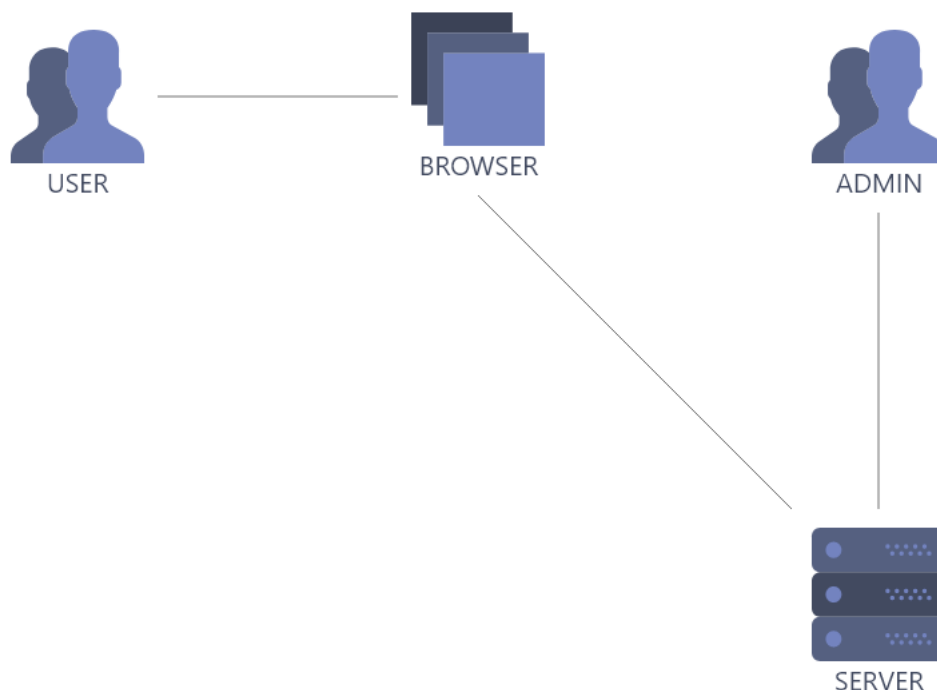


Рисунок 3 – JAM

Существует множество различных генераторов статических сайтов, один из которых используется в веб-ресурсе – это Gatsby [9]. Он совмещает использование React и GraphQL. Вероятно, выбор данной технологии был сделан, основываясь на статистике.

Contentful – это система управления контентом на статических страницах, которая отлично работает совместно с Gatsby и Netlify. Вместо создания связей между шаблоном и базой данных, она меняет данные в самих HTML файлах [10].

Netlify – это система, позволяющая создавать интуитивно понятный рабочий процесс на основе Git, используя мощную бессерверную платформу для создания, развертывания и совместной работы над веб-приложениями [11].

Исходя из используемых технологий в внедряемом веб-ресурсе, подберем наиболее подходящие библиотеки для реализации поставленных задач.

Исходя их поставленных в пункте 1.1 задач, можно сделать вывод, что необходимо разработать три независимо работающих сервиса: создание макета, рассылка писем на почту, оплата.

Для реализации первого сервиса, будет использоваться библиотека Konvajs [12]. Данная библиотека позволяет использовать canvasjs через декларативные компоненты библиотеки React. Использование чистого canvasjs нецелесообразно при наличии данного решения.

В качестве библиотек для отправки писем будет использоваться бесплатный Gmail API через библиотеку googleapis и nodemailer [13] для отправки электронных писем через SMTP протокол.

Так как необходимо реализовать сервис оплаты, то будет использоваться QIWI кошелек [14]. Кроме выбранного сервиса существует YooMoney, ранее известный, как Яндекс Деньги, разработанный компанией Яндекс, славящейся своей любовью к монетизации всего, к чему может прикоснуться. Так, например, за перевод на карту YooMoney, необходимо будет заплатить комиссию в 30 рублей, даже за перевод одного рубля. QIWI кошелек предлагает свои услуги бесплатно, например, комиссия за перевод на QIWI кошелек с карты Сбербанка составит 0%.

Сервис создания макета будет внедрен непосредственно в веб-ресурс, а оставшиеся сервисы необходимо будет разместить на отдельном сервере, который будет выступать в роли API. Сервер можно разместить на бесплатном хостинге Heroku.

В результате использования выбранных технологий, возможно будет достичь необходимой скорости в разработке поставленных задач.

Для непосредственной разработки будут использоваться следующие, зарекомендовавшие себя временем, средства:

					09.02.03.ИТ.ПДП.О	Лист
						12
Изм.	Лист	№ докум.	Подпись	Дата		

- Редактор кода VS Code
- Браузер Google Chrome

Редактор кода VS Code поддерживает разработку на множестве языков, в том числе и полную поддержку TypeScript, поскольку разработан той же компанией, что и язык.

Браузер Google Chrome является монополистом в мире браузеров. Он поддерживает все современные технологии веб-разработки и приносит новые.

Использование современных средств разработки позволяет разрабатывать и сопровождать качественные программные продукты.

Кроме вышеперечисленных средств, также необходимо система контроля версий Git. Для ее использования достаточно будет сервиса GitHub. Поскольку веб-ресурс, в который будет внедряться разработка, уже находится на репозитории GitHub, то разработка будет вестись в нем.

					09.02.03.ИТ.ПДП.О	Лист
						13
Изм.	Лист	№ докум.	Подпись	Дата		

## 2 СПЕЦИАЛЬНАЯ ЧАСТЬ

### 2.1 Моделирование диаграммы потоков данных

Перед непосредственной разработкой программных продуктов принято создавать диаграммы потоков данных. Для реализации поставленных задач необходимо создать такую диаграмму потоков данных, которая смогла бы в достаточной степени описать и структурировать данные с которыми взаимодействуют сущности в системе. На рисунке 4 отображена высокоуровневая диаграмма потоков данных. На представленном рисунке видно, что есть две сущности: клиент и администратор, один процесс: уточнение требований, данные: требования, оплата и заказ-макет, а также хранилище данных: заказ-макет.

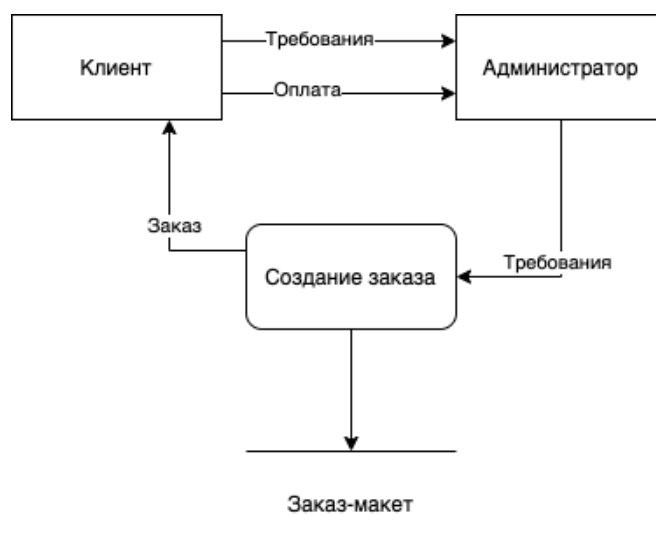


Рисунок 4 – Контекстная диаграмма потоков данных

При детализации процесса, имеющего проблему, можно представить её в виде, отображенном на рисунке 5. После детализации становится видно, что администратор занимается двумя делами вместо одного, в результате чего падает качество работы и увеличивается нагрузка на данную сущность.

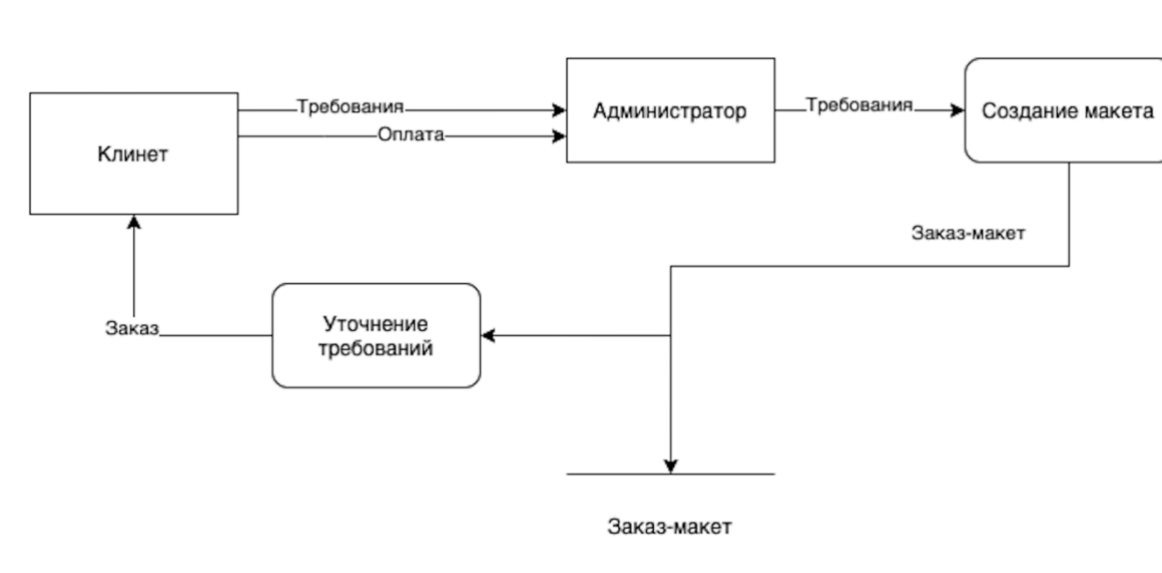


Рисунок 5 – Детализированная диаграмма потоков данных

Таким образом, после детализации стало видно, что предприятие имеет утечку производительности в области создания заказа. Данные действия можно возложить на клиента, предоставив ему удобные инструменты, благодаря которым, тот сможет сам создавать свои собственные макеты.

После внедрения разработки, администратору нужно будет лишь отвечать на письма заказов и передавать макеты на печать, тем самым он станет посредником между производителем и клиентом.

## 2.2 Проектирование моделей в Contentful CMS

В ContentfulCMS создаются модели контента. Для данной разработки спроектируем модель, необходимую для создания макета на стороне клиента.

Ниже представлен список с необходимыми полями:

- 1) Заголовок
- 2) Задний фон
- 3) Шаблон

- 4) Точка начала области печати по X
- 5) Точка начала области печати по Y
- 6) Высота области печати
- 7) Ширина области печати
- 8) Размер товара
- 9) Цена товара

Данную модель необходимо интегрировать в уже имеющуюся модель посредством связи один ко многим.

Таким образом после создания контента в Contentful CMS можно будет получить данные в корректном виде для отображения редактора макета и формы создания заказа в целом.

## 2.3 Проектирование веб-сервиса

Можно смоделировать диаграмму последовательностей, которая в достаточной степени отобразит последовательность процессов, необходимых для достижения поставленных задач. На рисунке 6 представлена диаграмма последовательностей веб-сервиса.

На рисунке 6 можно наблюдать 6 сущностей:

- Клиент
- Сервер
- БД (база данных)
- CMS
- QIWI
- EMAIL

Между представленными сущностями установлены взаимодействия, в результате которых пользователь может создать и оплатить заказ.

					09.02.03.ИТ.ПДП.О	Лист
						16
Изм.	Лист	№ докум.	Подпись	Дата		

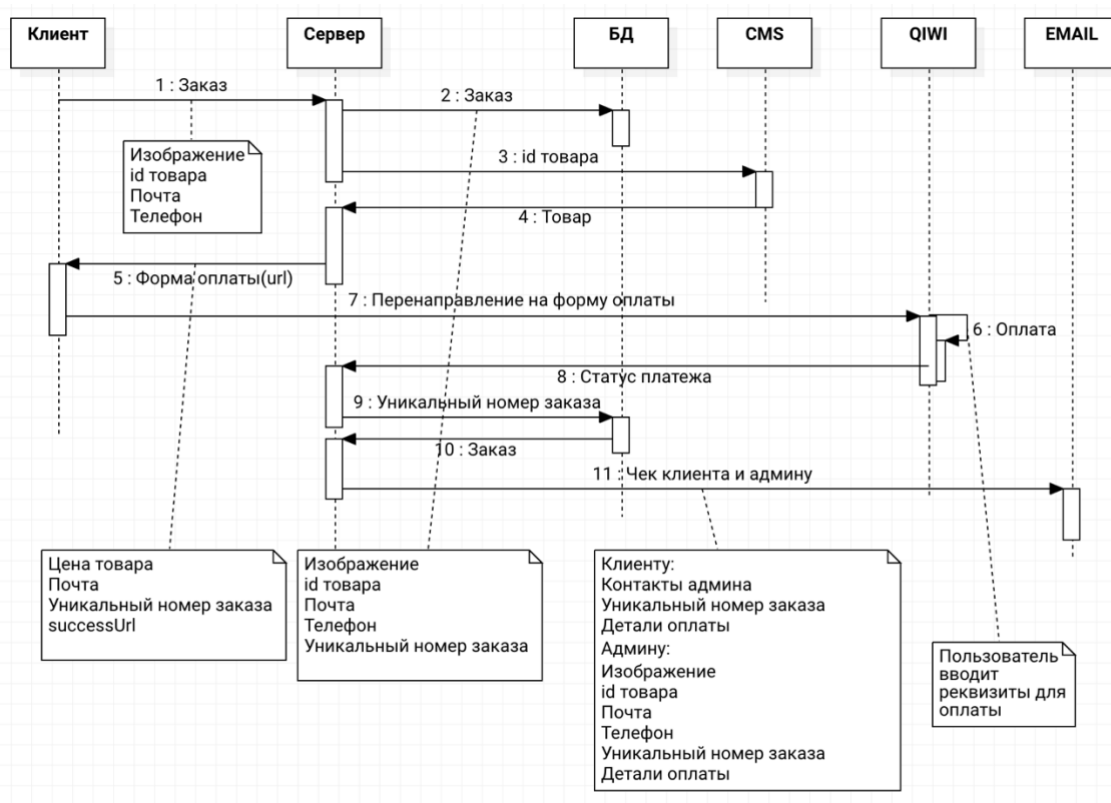


Рисунок 6 – диаграмма последовательностей веб-сервиса

## 2.4 Тестирование

При разработке веб-сервиса не выявилось ни одной ошибки, в результате чего, необходимость в отладке пропадает. Отсутствие ошибок вызвано использованием языка TypeScript, который в отличие от JavaScript является языком со статической типизацией, которая проверяется в момент компиляции. Таким образом, язык TypeScript не только позволяет избежать некоторых ошибок в runtime, но и ошибок в compile time, посредством вывода подсказок при написании кода, которые вызваны установкой типов для каждой функции, объекта или константы.



Однако ошибки могут возникнуть, поскольку TypeScript не является языком, который следит за типизацией в runtime, следовательно вследствие обращения к API могут возникнуть ошибки.

Ошибки могут отобразиться во время разработки в виде сообщения или некорректной работы. Зачастую сообщения об ошибках отображают место и причину, а иногда, вследствие особенностей языка JavaScript, они могут не отображаться, то есть быть неявными, которые можно исправить лишь используя Debugger.

Во избежание возможных ошибок, проводят различные тесты, например, unit-тестирование, тестирование методом белого или черного ящика.

Для тестирования разрабатываемого веб-сервиса выбран метод тестирования черного и белого ящика, поскольку время на написание Unit-тестов отсутствует. При тестировании достигается необходимая гарантия того, что веб-сервис будет работать корректно при различных условиях.

Метод черного ящика заключается в тестировании, не предполагающим знания внутреннего устройства системы. Благодаря данному тестированию можно не только сказать, что система работает верно, но и то, что пользователь сможет с легкостью разобраться в ней.

Тестирование методом черного ящика проводилось между закрытым кругом лиц, являющимися добровольцами в прохождении тестирования. В таблице 1 отображены результаты тестирования методом черного ящика.

Таблица 1 – результаты теста методом черного ящика

Описание теста	Ожидаемый результат	Результат
Открытие редактора макета	Открытый редактор макета	Пройдено
Выбор макета через селектор	Выбранный макет	Пройдено
Добавление текста на макет	Отображение текста на макете	Пройдено
Добавление нескольких изображений на макет	Отображение изображений на макете	Пройдено

Описание теста	Ожидаемый результат	Результат
Перемещение текста	Перемещенный текст	Пройдено
Перемещение изображений	Перемещенное изображение	Пройдено
Изменение цвета текста	Измененный цвет текста	Пройдено
Изменение размера текста	Измененный размер текста	Пройдено
Изменение размера изображений	Измененный размер изображений	Пройдено
Ввод email, телефона и имени в форму	Введенные email, телефон и имя	Пройдено
Оплата используя карту или qiwі кошелек	Оплата используя карту или qiwі кошелек	Пройдено
Получение письма на почту администратором с изображениями, товаром, email, телефоном, именем	Полученное письмо на почте администратора	Пройдено
Получение письма с чека на почту клиента	Получение письма с чеком на почте клиента	Пройдено

Тестирование методом белого ящика будет использоваться для сравнения требований, описанных в техническом задании, и реализованного функционала. В таблице 2 отображены результаты тестирования методом белого ящика.

Таблица 2 – результаты теста методом белого ящика

Описание теста	Ожидаемый результат	Результат
Открытие редактора макета	Открытый редактор макета	Пройдено
Выбор макета через селектор	Выбранный макет	Пройдено
Добавление текста на макет	Отображение текста на макете	Пройдено
Добавление нескольких изображений на макет	Отображение изображений на макете	Пройдено

Описание теста	Ожидаемый результат	Результат
Перемещение текста	Перемещенный текст	Пройдено
Перемещение изображений	Перемещенное изображение	Пройдено
Изменение цвета текста	Измененный цвет текста	Пройдено
Изменение размера текста	Измененный размер текста	Пройдено
Изменение размера изображений	Измененный размер изображений	Пройдено
Ввод email, телефона и имени в форму	Введенные email, телефон и имя	Пройдено
Оплата используя карту или qiwi кошелек	Оплата используя карту или qiwi кошелек	Пройдено
Получение письма на почту администратором с изображениями, товаром, email, телефоном, именем	Полученное письмо на почте администратора	Пройдено
Получение письма с чека на почту клиента	Получение письма с чеком на почте клиента	Пройдено

По результатам тестов можно сказать, что веб-сервис работает согласно требованиям и готов к внедрению.

## 3 ЭКОНОМИЧЕСКАЯ ЧАСТЬ

### 3.1 Расчет себестоимости

Существуют множество техник и подходов в оценке проектов. Основных моделей оценки два, которые тесно связаны с основными подходами к управлению. Так, например, в статье Виталия Салахмира, являющимся опытным руководителем проектов, про оценку стоимости разработки проекта в достаточной полной форме описаны данные модели [17].

Ниже представлены две основные модели оценки проекта:

- 1) Fix-price
- 2) Time and material

По модели оценки Fix-price производится оценка сроков и стоимости всего проекта целиком. Зачастую данная модель используется тогда, когда существуют строго определенное техническое задание.

При использовании модели Time and material оценка сроков и стоимости происходит во время разработки.

Согласно модели Time and material, оценка происходит путем выяснения того, сколько нужно времени на разработку и умножением его на стоимость часа работы специалиста.

Оценка стоимости проекта по модели Fix-price происходит один раз, таким образом заказчик экономит время на операциях, связанных с оценкой каждой задачи, а также заплатить больше или заплатить меньше, чем стоит проект. Однако в некоторых случаях оценка по данной модели может пересматриваться, так как угадать реальную цену зачастую не удастся, и тогда необходимо повторное оценивание, но обычно, обсуждается лишь добавочное время, нежели доплата.

					09.02.03.ИТ.ПДП.О	Лист
						21
Изм.	Лист	№ докум.	Подпись	Дата		

Модель Time and material предполагает оплату за затраченное время, например, при гибкой разработке получение оплаты происходит после выполнения каждой задачи или же перед выполнением задачи производится оценка и оплата. Данная модель позволяет оценивать стоимость конкретных задач. Оценка стоимости каждой задачи точно также является задачей, имеющая свою стоимость. При данном подходе производятся исследования более явно, чем при Fix-price.

Разработка настоящего веб-сервиса является бесплатной, однако оценивается по модели Time and material, поскольку она в достаточной степени позволит оценить выполненную работу.

Согласно выбранной модели оценки, она может производиться менеджером проекта, тим-лидом, руководителем отдела или непосредственным исполнителем. Таким образом, оценка производится непосредственным исполнителем, ввиду отсутствия вышеперечисленных.

При оценке непосредственным исполнителем достигается максимальная точность, потому что он разбирается в нюансах задачи, а также именно он будет её реализовывать и нести ответственность за выполнение.

Поскольку разработчик настоящего веб-сервиса является достаточно опытным, чтобы считать его Middle-разработчиком, поэтому оценка будет вестись с учётом профессионального уровня специалиста.

### 3.2 Определение трудоемкости

Выделим используемых для реализации проекта сотрудников:

- Программист
- Дизайнер
- Верстальщик

					09.02.03.ИТ.ПДП.О	Лист
						22
Изм.	Лист	№ докум.	Подпись	Дата		

- Тестировщик
- Менеджер

В таблице 3 отображены потраченные часы и стоимость часа каждого сотрудника.

Таблица 3 – Стоимость проекта

Сотрудник	Часы	Стоимость часа	Стоимость
Программист	38	800	30400
Дизайнер	8	100	800
Верстальщик	8	300	2400
Тестировщик	2	100	200
Менеджер	24	500	12000

Время, затраченное на разработку, равняется 10 дням по 8 часов работы в день.

Поскольку работа велась в бесплатном выделенном помещении, используя бесплатную электроэнергию, то стоимость часа исходит исключительно из чистой работы без надбавок, однако в стоимость часа включаются затраты на оборудование.

Цена за час взята, как средняя по рынку веб-разработки, учитывая опыт и навыки сотрудника.

### 3.2.1 Расчет материальных затрат

Материальные затраты исходят из затрат на используемое оборудование, а также иные нужды, какие укажет разработчик в разумных пределах. Например, затраты на питание по средней цене в столовых.

Вышеперечисленные затраты отображены в таблице 4.

Таблица 4 – Расчет материальных затрат

Материальные затраты	Стоимость затрат в день в рублях
MacBook Pro	10
Иные нужды	250

Стоимость затрат на MacBook Pro рассчитывалась как стоимость замены батареи, деленная на гарантийный срок службы. Иные нужды указываются как средние затраты в день. Таким образом стоимость материальных затрат за всё время разработки будет 2150 рублей.

### 3.2.2 Расчет затрат на электроэнергию и амортизации оборудования

Поскольку электричество во время разработки являлось бесплатным, то затраты на него не учитываются при вычислении финальной цены разработки. Однако при платной электроэнергии, расчет производился бы путем умножения стоимости электричества в час на количество выработанных часов программистом, дизайнером, верстальщиком, тестировщиком и менеджером.

Расчет амортизации оборудования ---описать окупаемость ноута---

### 3.2.3 Расчет расходов на заработную плату

Подведем итоги по заработной плате согласно таблице 3, в которой указана стоимость работы сотрудника в час. В таблице 4 приведена заработная плата в рублях для каждого сотрудника, принимающего участие в проекте.

Таблица 4 – Заработная плата

Сотрудник	Заработная плата в рублях
-----------	---------------------------

Программист	30400
Дизайнер	800
Верстальщик	2400
Тестировщик	200
Менеджер	12000
Итого: 45800	

**ИТОГО**

### 3.2.4 Расчет себестоимости и цены разработки

Согласно проведенным вычислениям, просуммируем все затраты на разработку. Будем использовать формулу:

$$CP = (ЗП * 0,1) + ЗП + МЗ \quad (1)$$

где CP – стоимость разработки в рублях;

ЗП – заработная плата;

МЗ – материальные затраты.

Подставив в формулу значения, рассчитанные выше, учитывая наценку фирмы в 10% получим:

$$CP = (45800 + 2100) * 0,1 + 45800 + 2100 = 52690 \text{ рублей}, \quad (2)$$

По итогам расчета, финальная цена разработки составила 52690 рублей.

### 3.2.5 Расчет экономического эффекта

**Сравнить то что сейчас и то что будет после внедрения. Сколько заказов будет. На сколько процентов увеличится прибыль. Показать реальные сроки**



## 4 ОБЕСПЕЧЕНИЕ БЕЗОПАСНОСТИ

### 4.1 Разработка мер защиты информации от несанкционированного доступа

В разрабатываемом веб-сервисе присутствует сервис оплаты, который обязан поддерживать передачу данных по протоколу HTTPS. Оплата происходит максимально безопасным методом, через заведомо надежный сервис онлайн платежей QIWI Кошелек.

При заполнении заявки пользователем на клиенте, данные отправляются на сервер, который отправляет пользователя на страницу оплаты, созданную QIWI Кошельком. После успешной или безуспешной оплаты, QIWI Кошелек отправляет чек об оплате на секретный эндпоинт сервера.

Таким образом, можно с уверенностью заявить, что оплата на разрабатываемом веб-сервисе безопасна. Также стоит отметить, что данные пользователя, указанные им при составлении заявки, хранятся на сервере лишь до окончания оплаты, тем самым уменьшается риск утечки данных.

### 4.2 Мероприятия по обеспечению безопасности жизнедеятельности

Во время разработки необходимо не только задумываться о безопасности реализуемого проекта, будь то архитектурное решение или написанный код, также, что более важно, необходимо уделять внимание безопасности жизнедеятельности при работе за компьютером.

В общем доступе через глобальную сеть интернет представлено множество техник, обеспечивающих безопасность жизнедеятельности.

					09.02.03.ИТ.ПДП.О	Лист
						26
Изм.	Лист	№ докум.	Подпись	Дата		

Согласно множеству источников, ниже представлены шесть основных опасных и вредных факторов:

- 1) Повышенный уровень электромагнитных излучений
- 2) Повышенный уровень статического электричества
- 3) Пониженная ионизация воздуха
- 4) Физические перегрузки в статическом положении
- 5) Перенапряжение глаз
- 6) Недостаточная освещенность

Исходя из представленных вредных и опасных факторов, необходимо организовать мероприятия по уменьшению воздействия данных факторов.

Например, в начале работы необходимо производить настройку и регулировку рабочего места для уменьшения нагрузки в статическом положении.

Во время работы важно делать физические разминки глаз и тела, а также проветривать помещения, и иметь ионизаторы воздуха.

Недостаточную освещенность можно предотвратить наличием прямого доступа солнечных лучей на рабочее место, однако излишняя освещенность может навредить. При работе в темное время суток, устанавливаются специальные лампы, создающее фоновое освещение с мощностью от 40 до 60 Вт, например, светодиодные и галогенные лампы. Желательно располагать лампы под монитором на высоте около 35 см [19].

По окончании работы обязательным мероприятием должно быть приведение рабочего места в изначальное состояние, выключаются рабочие устройства и используемые предметы убираются на место. Таким образом, уменьшаются излучения в момент бездействия.

Разработка велась с учетом вышеперечисленных правил. При работе за персональным компьютером соблюдались правила расположения относительно стола и монитора. Во время работы и перерыва был включен озонатор воздуха, проветривались помещения, а также, в темное время суток,

					09.02.03.ИТ.ПДП.О	Лист
						27
Изм.	Лист	№ докум.	Подпись	Дата		

использовалось фоновое дневное освещение. В перерывах между работой проводилась тренировка, представленная на YouTube канале Dev Ed, имеющем 647 тысяч подписчиков [20].

Исходя из проводимых мероприятий по обеспечению безопасности жизнедеятельности при работе за компьютером, можно сказать, что здоровье разработчика сохраняется успешно.

## ЗАКЛЮЧЕНИЕ

Согласно пункту 1.1, цель сформулирована следующая: «Распределить нагрузку на предприятии за счет веб-сервиса создания и оплаты заказов онлайн без участия сотрудников». По результатам выполненных работ, можно утверждать, что проект готов к внедрению на предприятие, поставленная цель достигнута.

Внедренный веб-сервис позволит предприятию увеличить качество производимой продукции за счет составления заказа онлайн, а именно, создание макета и заполнение контактных данных, а также его последующей оплаты, что значительно снизит нагрузку на администратора. Увеличение качества продукции и обслуживания увеличивает клиентопоток, что в свою очередь, увеличивает прибыль компании. **сослаться на эконом эффективность**

Предоставление клиенту возможности составлять заказы удаленно, повысит уровень доверия к предприятию, что увеличит процент повторных заказов. Реализовывая свой творческий потенциал, клиенты смогут создавать макеты в удобном месте, совместно со своими знакомыми или родными.

За счет использования Contentful CMS, администратор может добавлять столько макетов, сколько необходимо, что является огромным преимуществом, за счет своей цены и скорости добавления.

После внедрения возможно дальнейшее развитие разработки, поскольку она велась с использованием гибких методологий, что позволило сохранить гибкую архитектуру. Возможно добавление нового функционала в создание макета, а также добавление различных способов оплаты, например, Apple Pay и Google Pay.

					09.02.03.ИТ.ПДП.О	Лист
Изм.	Лист	№ докум.	Подпись	Дата		29

## СПИСОК ИСПОЛЬЗОВАННЫХ ИСТОЧНИКОВ

1 Условия повышения качества продукции [Электронный ресурс] – Режим доступа: <http://www.stroitelstvo-new.ru/sudostroenie/brigadir/povyshenie-kachestva-produktsii.shtml> (дата обращения 26.05.21).

2 Сервис печати [Электронный ресурс] – Режим доступа: <https://www.vsemayki.ru/> (дата обращения 27.05.21).

3 Agile, scrum, kanban: в чем разница и для чего использовать? [Электронный ресурс] — Режим доступа: <https://rb.ru/story/agile-scrum-kanban> (Дата обращения: 08.05.21)

4 Что такое Agile? Канбан – это Agile? [Электронный ресурс] — Режим доступа: <https://rb.ru/story/kanban-agile/> (Дата обращения: 27.05.21)

5 Цитаты известных личностей [Электронный ресурс] — Режим доступа: <https://ru.citaty.net/tsitaty/472306-klod-adrian-gelvetsii-znanie-nekotorykh-printsipov-legko-vozmeshchaet-neznani/> (Дата обращения: 27.05.21)

6 Методологии разработки [Электронный ресурс] — Режим доступа: <https://issoft.by/blog/podkhody-k-razrabotke-po-kak-pravilno> (Дата обращения: 30.05.21)

7 Что такое Jamstack [Электронный ресурс] — Режим доступа: <https://jamstack.org/what-is-jamstack/> (Дата обращения: 30.05.21)

8 Jamstack [Электронный ресурс] — Режим доступа: <https://habr.com/ru/post/439232/> (Дата обращения: 30.05.21)

9 Gatsby [Электронный ресурс] — Режим доступа: <https://www.gatsbyjs.com/>

10 Contentful [Электронный ресурс] — Режим доступа: <https://www.contentful.com/>

11 Netlify [Электронный ресурс] — Режим доступа: <https://www.netlify.com/> (Дата обращения: 30.05.21)

					09.02.03.ИТ.ПДП.О	Лист
						30
Изм.	Лист	№ докум.	Подпись	Дата		

- 12 Konvajs <https://konvajs.org/>
- 13 Nodemailer <https://nodemailer.com/about/>
- 14 QIWI <https://qiwi.com/>
- 15 Heroku <https://heroku.com>
- 16 JS Fest 2019 [Электронный ресурс] — Режим доступа:  
<https://www.youtube.com/watch?v=wCNSK4iFCuE>
- 17 <https://salakhmir.ru/оценка-стоимости-разработки-проекта>
- 18 <https://www.kadrof.ru/articles/46641>
- 19 <https://wd-x.ru/osveshhenie-pri-rabote-za-kompyuterom/>
- 20 <https://www.youtube.com/watch?v=qL1YjIRDQEI>

**ПРИЛОЖЕНИЕ А**  
**ПРИЛОЖЕНИЕ В**  
**ПРИЛОЖЕНИЕ С**



