

# Доклад

---

Тема “Проектирование и разработка веб-сервиса приема и оплаты заказов на примере сервисного центра «Папа-Принтер»”.

## 2

---

### Востребованность

---

Последнее время, в связи с развитием цифровой экономики и технологий, любому предприятию без представления себя в интернете, не обойтись. В связи с этим, разработка веб-сервисов для предприятий актуальна и востребована.

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

Проблема исходит из реальных неудобств и жалоб клиентов, которые приходили на предприятие и хотели сразу получить желаемый товар. Однако, прежде чем его получить, необходимо было сначала создать макет для печати, поэтому получение заказа откладывалось.

## 3

---

### Цели разработки

---

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

## 4

---

### Методология разработки

---

Исходя из сформулированной цели разработки, можно сделать вывод, что наиболее подходящей методологией разработки является...

## 5

---

...гибкая методология с использованием двух подходов.

- Scrum – для разделения разработки на рабочие части;
- Kanban – для визуализации рабочего процесса.

## 6

---

### JAMstack

---

Веб-ресурс разрабатывался, с использованием подхода JAMstack.

Данный подход подразумевает разработку веб-приложения на желаемом фреймворке и управление контентом из CMS. В результате получатся HTML страницы, которые были сгенерированы с использованием рекомендаций SEO, PWA и Accessibility (A11y).

JAM подход предполагает отказ от рендеринга на стороне сервера и клиента, компиляцию всего сайта в готовые HTML страницы единой порцией при изменении контента, а также размещение на практически любом файловом хостинге. Таким образом, при грамотной разработке, производительность измеряется только в скорости интернет-соединения пользователя.

## Технологии для JAM

Существует множество различных генераторов статических сайтов, один из которых используется в веб-ресурсе – это Gatsby. Он совмещает использование React и GraphQL.

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. Говоря простым языком – это хостинг с возможностью запуска скриптов.

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 позволяет менять данные в самих HTML файлах.

# 7

## Технологии и средства для реализации поставленной цели

### Конструктор (Konvajs)

Данная библиотека позволяет использовать canvasjs через декларативные компоненты библиотеки React. Использование чистого canvasjs нецелесообразно при наличии данного решения.

### Отправка почты (Nodemailer)

Для отправки электронных писем используют SMTP протокол. Для взаимодействия с ним, используется данная библиотека. Она поддерживает Gmail API, который и предоставляет возможность отправлять письма бесплатно.

### QIWI кошелек

QIWI кошелек предлагает свои услуги бесплатно. Заведомо безопасен и имеет удобное и простое API.

### Редактор кода (VS Code)

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 привносит новые.

## GitHub

Кроме ранее перечисленных 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.

## 8

## Проектирование

Перейдем к тому как проектировался веб-сервис.

## 9

## Моделирование диаграммы потоков данных

На контекстной диаграмме вроде бы всё отлично.

Однако после детализации стало видно, что предприятие имеет утечку производительности в области создания заказа. Администратор занимается лишней работой.

## Для решение проблемы

Создание макета можно возложить на клиента, предоставив ему удобные инструменты, благодаря которым, тот сможет сам создавать свои собственные макеты.

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

## 10

## Создана модель в Contentful

1. Заголовок
2. Шаблон
3. Область печати
4. Размер товара
5. Цена товара

## 11

## Для проектирование веб-сервиса была составлена

Диаграмма последовательностей, в достаточной степени отображающая последовательность процессов, которые необходимы для достижения поставленных задач.

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

## 12

---

### Тестирование и отладка

---

#### Отладка

При разработке веб-сервиса не выявилось ни одной ошибки, в результате чего, необходимость в отладке пропадает. Отсутствие ошибок вызвано наличием опыта и использования языка TypeScript, который в отличие от JavaScript является языком со статической типизацией, которая проверяется в compile time.

#### Тестирование

Тестирование методом черного ящика проводилось между закрытым кругом лиц, являющимися добровольцами в прохождении тестирования.

А метод белого ящика использовался для сравнения требований, описанных в техническом задании, и реализованного функционала.

## 13

---

### Экономическая часть

---

#### Основные модели оценки проекта

Существуют множество техник и подходов в оценке проектов.

Основных моделей оценки два, которые тесно связаны с основными подходами к управлению.

#### Fix-price

Оценка стоимости проекта происходит один раз.

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.

## Time and material

Предполагает оплату за затраченное время.

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.

## Выбранная модель

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
- Непосредственным исполнителем

Оценка производилась 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 велась с учётом профессионального уровня специалиста.

## Стоимость проекта

По итогам расчета, финальная цена разработки с учетом всех ее составляющих расходов составила 54 тысячи рублей.

## Экономический эффект

Предположительно, через 6 месяцев предприятие увеличит клиентопоток на 60%, а количество заказов на 360%.

## Защита веб-сервиса

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, которые он указал при составлении заявки, хранятся на сервере лишь до окончания оплаты, тем самым уменьшается риск утечки данных.

## Безопасность жизнедеятельности

Во время разработки необходимо не только задумываться о безопасности реализуемого проекта, но и, что более важно, необходимо уделять внимание безопасности при работе за компьютером.

При работе за соблюдались правила расположения относительно стола и монитора.

Во время работы и перерыва был включен озонатор воздуха, проветривались помещения, а также, в темное время суток, использовалось фоновое дневное освещение.

В перерывах между работой проводилась разминка.

## 15

## Заключение

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.

## 16

Кто желает, пожалуйста подготовьте смартфоны, чтобы отсканировать QR-код для перехода на сайт.

## 17

Прежде чем перейдем к демонстрации, обратите внимания на тесты производительности. Скорость загрузки составляет 2 десятых секунды.

## 18

Это доказывает, что использование JAMstack позволяет добиться того, чтобы скорость загрузки сайта зависла исключительно от скорости интернета.

## Перейдем к демонстрации разработки

---

Показать:

- Сайт в целом – на главной странице два раздела, показать фотоновик
- Выбор шаблона – в разделе фотоновика есть конструктор макета
- Выбор размера товара
- Добавление текста на шаблон, изменение его цвета, размера, местоположения и области
- Добавления изображения, изменение размера и местоположения
- Удаление текста
- Удаление изображения
- Добавить еще одно фото
- Заполнение формы заказа
- Оплата
- Красивая страница успешной оплаты
- Письма на почте администратора и клиента

## На этом мое выступление окончено!

---