

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	13
2.2 Проектирование базы данных	14
2.3 Проектирование программного обеспечения	15
2.4 Тестирование и отладка	15
3 ЭКОНОМИЧЕСКАЯ ЧАСТЬ	18
3.1 Расчет себестоимости.....	18
3.2 Определение трудоемкости	18
3.2.1 Расчет материальных затрат	18
3.2.2 Расчет затрат на электроэнергию амортизации оборудования	18
3.2.3 Расчет расходов на заработную плату	18
3.2.4 Расчет себестоимости и цены разработки	18
3.2.5 Расчет экономического эффекта	18
4 ОБЕСПЕЧЕНИЕ БЕЗОПАСНОСТИ.....	18
4.1 Разработка мер защиты информации от несанкционированного доступа.....	18
4.2 Мероприятия по обеспечению безопасности жизнедеятельности	18
ЗАКЛЮЧЕНИЕ.....	18

					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 ИСТОЧНИКОВ18

ПРИЛОЖЕНИЕ А.....18

ПРИЛОЖЕНИЕ В.....19

ПРИЛОЖЕНИЕ С.....19

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 «vsemaykiru», 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 «vsemaykiru» 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 методологий.

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 A11y.

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

Александр Товмач в своей публикации на портале Nabr предоставил сравнение трех подходов в веб-разработке, которые будут кратко представлены ниже [8].

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.

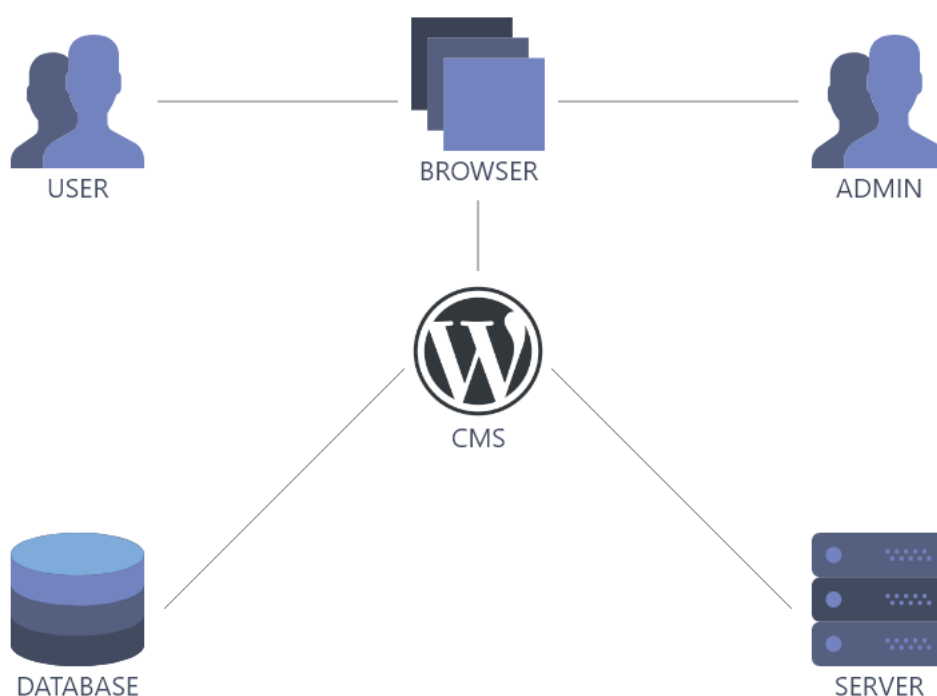


Рисунок 1 – PHP сервер-рендеринг

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.

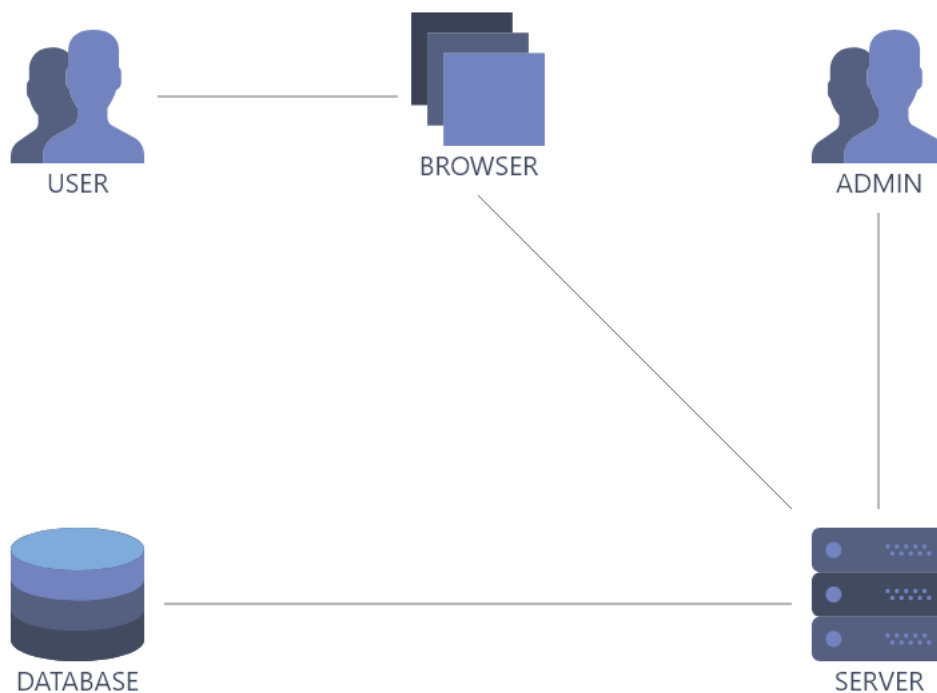


Рисунок 2 – Подход с использованием React, Vue или Angular

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.

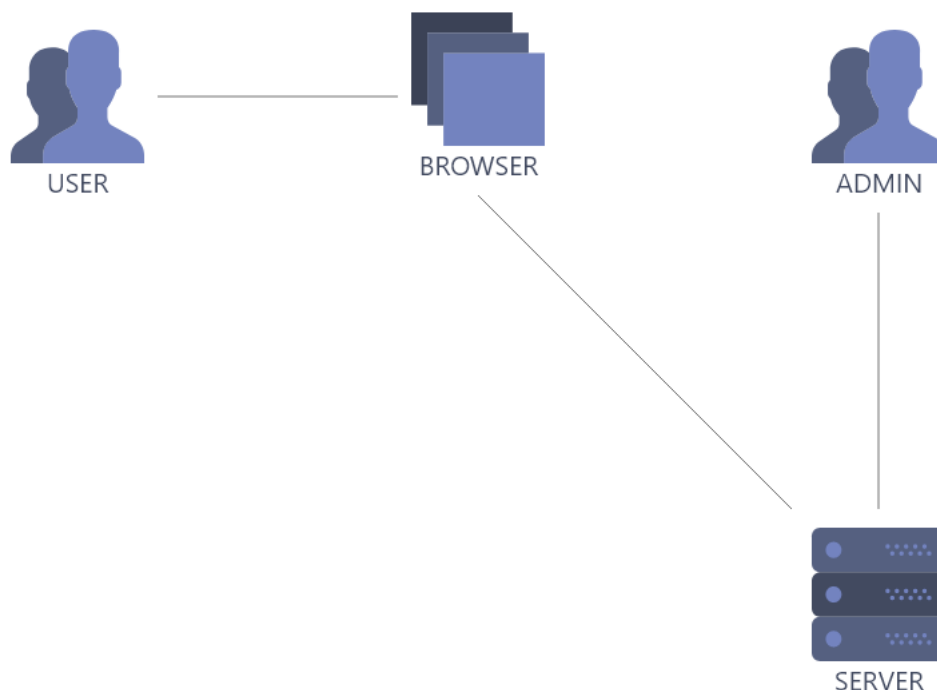


Рисунок 3 – JAM

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 из поставленных в пункте 1.1 задач, можно сделать вывод, что необходимо разработать три независимо работающих сервиса: создание макета, отправка 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 GmailAPI через библиотеку googleapis и nodemailer [13] для отправки электронных писем через SMTP протокол компании google.

Так как необходимо реализовать сервис оплаты, то будет использоваться QIWI кошелек [14]. Кроме выбранного сервиса существует YooMoney, ранее известные 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%, как и обратно.

Первый сервис будет внедрен непосредственно в веб-ресурс, а вторые две необходимо будет разместить на отдельном сервере, который будет выступать в роли API. Сервер можно разместить на бесплатном хостинге Heroku.

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

					09.02.03.ИТ.ПДП.О	Лист
						12
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 в достаточной степени описать и структурировать данные, 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 – Высокоуровневая 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.

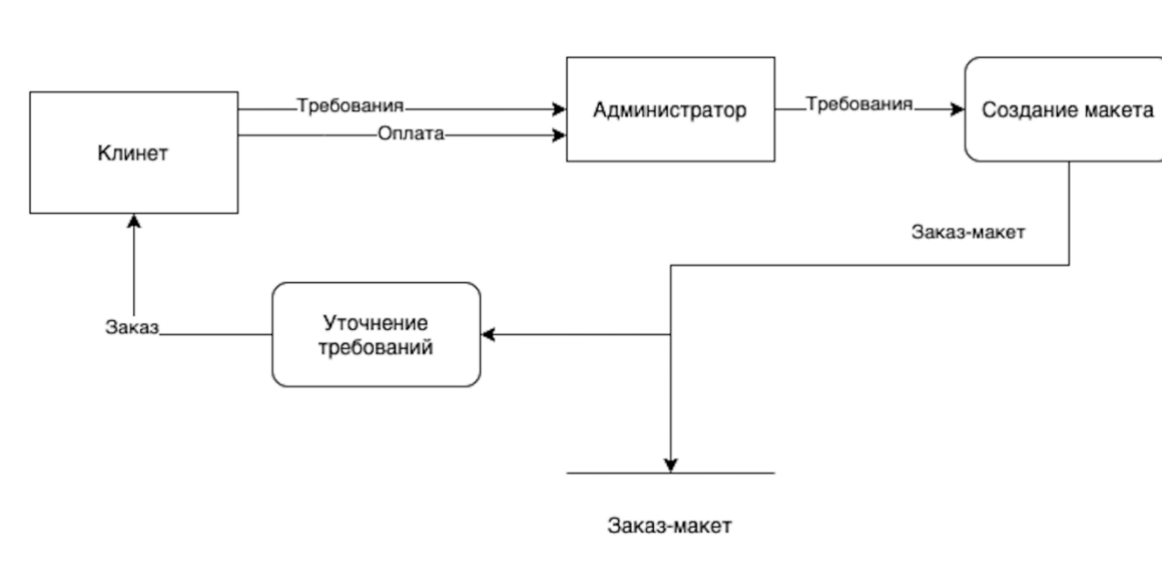


Рисунок 5 – Низкоуровневая диаграмма потоков данных

Таким образом, после детализации стало видно, что предприятие имеет утечку производительности в области создания макета и уточнения требований. 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

Опишу позже, потому что там много (средне).

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.

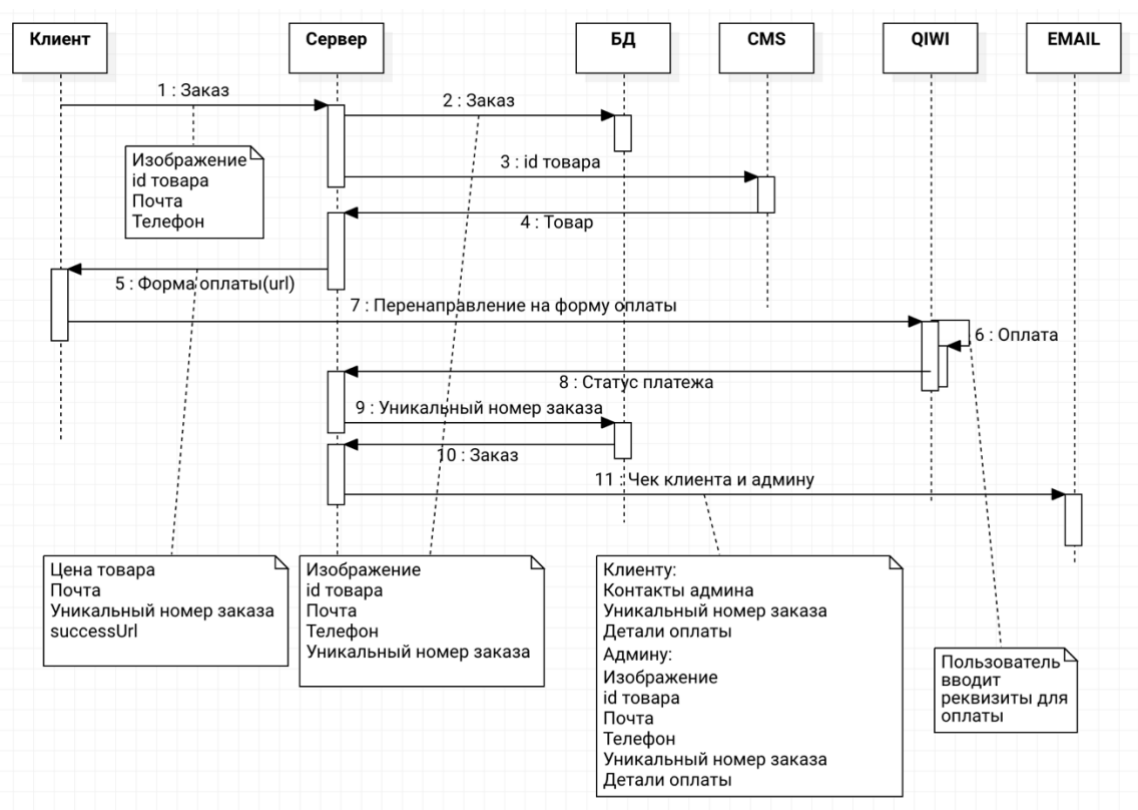


Рисунок 6 – диаграмма последовательностей веб-сервиса

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 времени на написание Unit-тестов отсутствует.

Метод черного ящика заключается в...

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

Тестирование методом черного ящика проводилось...

Метод белого ящика заключается в...

Тестирование методом белого ящика проводилось...

По результатам теста можно сказать...

					09.02.03.ИТ.ПДП.О	Лист
						17
Изм.	Лист	№ докум.	Подпись	Дата		

3 ЭКОНОМИЧЕСКАЯ ЧАСТЬ

3.1 Расчет себестоимости

3.2 Определение трудоемкости

3.2.1 Расчет материальных затрат

3.2.2 Расчет затрат на электроэнергию амортизации оборудования

3.2.3 Расчет расходов на заработную плату

3.2.4 Расчет себестоимости и цены разработки

3.2.5 Расчет экономического эффекта

4 ОБЕСПЕЧЕНИЕ БЕЗОПАСНОСТИ

4.1 Разработка мер защиты информации от несанкционированного доступа

4.2 Мероприятия по обеспечению безопасности жизнедеятельности

ЗАКЛЮЧЕНИЕ

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)

12 Konvajs <https://konvajs.org/>

13 Nodemailer <https://nodemailer.com/about/>

14 QIWI <https://qiwi.com/>

15 Heroku <https://heroku.com>

ПРИЛОЖЕНИЕ А
ПРИЛОЖЕНИЕ В
ПРИЛОЖЕНИЕ С

					09.02.03.ИТ.ПДП.О	Лист
						19
Изм.	Лист	№ докум.	Подпись	Дата		

