

Министерство образования и науки Пермского края
Государственное бюджетное профессиональное образовательное учреждение
«Осинский колледж образования и профессиональных технологий»

ВЫПУСКНАЯ КВАЛИФИКАЦИОННАЯ РАБОТА

РАЗРАБОТКА ВЕБ-САЙТА ДЛЯ РЕКЛАМНОГО АГЕНТСТВА «ROCONT»

Погадаев Вячеслав Олегович

Специальность 09.02.04

Информационные системы (по отраслям)

Курс IV, группа ИС-15-4

Научный руководитель

Евдокимова Ксения Михайловна

«Допустить к защите»

Заведующий учебной частью

_____ О.С.Юрганова

г. Оса, 2019

Содержание

Введение.....	3
1 Аналитическая часть.....	5
1.1 Техническо-экономическая характеристика компании.....	5
1.2 Формулировка задачи.....	8
1.3 Современные технологии, используемые для разработки веб-сайта.....	9
1.3.1 Методы разработки веб-сайтов.....	9
1.3.2 Проектирование веб-дизайна.....	16
1.4 Этапы создания веб-сайта.....	19
1.5 Программное обеспечение, необходимое для создания веб-сайта.....	20
1.5.1 Текстовые редакторы.....	20
1.5.2 Графические редакторы.....	21
1.5.3 Локальные сервера.....	21
2. Разработка веб-сайта.....	23
2.1 Разработка прототипа веб-сайта.....	23
2.2 Создание дизайна веб-сайта.....	33
2.3 Верстка веб-сайта.....	37
Заключение.....	40
Список используемых источников.....	42

Введение

В наше время интернет используется как инструмент удаленной коммуникации, который позволяет людям обмениваться различной информацией. За последние годы интернет оказал огромное влияние на развитие компаний, изменяя способы продвижения перед потенциальными клиентами и уровнем обслуживания действующих клиентов. Интернет помогает в росте бизнеса своей масштабностью, как крупным игрокам рынка, так и начинающим предпринимателям. В интернете есть возможности для автоматизации бизнес-процессов. Количество людей, использующих интернет, как место покупки товара или получения услуги значительно растет.

Чтобы представлять свои услуги в интернете необходимо обзавестись площадкой, оптимальной для этой задачи будет веб-сайт. Он является набором взаимосвязанных страниц, сгруппированных по разделам содержащих, текстовую и графическую информацию. Веб-сайт может, является визиткой компании для получения информации, в какой сфере работает компания и для получения контактных данных. Наличие своего веб-сайта, повышает доверительное отношение к компании. Также веб-сайт помогает увеличить конверсию продаж, решает главные маркетинговые задачи и автоматизирует работу, офисных сотрудников, такую как ответы на вопросы о местоположении, наличие товара, списка услуг. Также веб-сайт значительно помогает взаимодействию клиента с компанией, за счет форм обратной связи для онлайн-заявок.

Появление технологии создание веб-сайтов, заставила переосмыслить работу человека не только с информацией, но и с компьютером в целом. В связи с разработкой технологии веб в интернете начало появляться большое количество веб-сайтов, предоставляющих свои услуги. По моему мнению, тема является актуальной, так как в сети интернет с каждым днем подключается все больше и больше предпринимателей, следовательно,

разработка веб-сайта рекламного агентства, поможет привлечь новых заказчиков и автоматизировать бизнес-процессы.

Из всего вышеперечисленного следует, что веб-сайт является необходимым инструментом для развития современной компании и определяет тему выпускной квалификационной работы, разработка веб-сайта для рекламного агентства «Rocont».

Объектом исследования выпускной квалификационной работы является, рекламное агентство «Rocont» — международное рекламное агентство полного цикла, специализация компании, комплексное продвижение площадок в интернете и в жизни.

Предметом исследования является, технологии создания, методы разработки и продвижения веб-сайта.

Цель выпускной квалификационной работы, является проектирование и разработка веб-сайта, автоматизация бизнес-процессов, приобретение теоретических и практических знаний, с применением различных технологий.

Для реализации проекта необходимо выполнить следующие задачи:

1. Проанализировать деятельность компании «Rocont»;
2. Исследовать современные методы разработки веб-сайта;
3. Разработать веб-дизайн;
4. Разработать верстку веб-сайта.

1. Аналитическая часть

1.1 Техническо-экономическая характеристика компании

Данная компания, выбранное место в качестве базы для исследования выпускной квалификационной работы, было обусловлено необходимостью введения автоматизации бизнес-процессов на предприятии. Развитие деятельности компании, на рынке услуг, продвижения веб-сайтов.

Официальное наименование компании «Roscont», полное наименование «RoscontAdvertising», компания является международным рекламным агентством полного цикла, занимающееся онлайн и оффлайн продвижением. Специализация компании комплексное продвижение в интернете, брендинг и оффлайн реклама. Компания образовалась, на рынке рекламного продвижения в 2018 году, ориентируясь на продвижение современных веб-сайтов. Затем простая студия переросла в рекламное агентство с множеством услуг. Сейчас компания занимает высокие позиции по городу Санкт-Петербург, среди аналогичных компаний.

Основной деятельностью является:

1. разработка веб-сайтов;
2. реклама в интернете;
3. дизайн социальных сетей, веб-дизайн;
4. 3D визуализация;
5. разработка программного обеспечения;
6. фото и видеосъемка.

В компании работают, более 32 сотрудников. Разработкой веб-сайтов и дальнейшее их продвижением занимается 9 человек, чем больше лиц занимается проектом, тем разностороннее он получается. Слаженная работа — это залог успеха и контроля на каждом этапе. Для каждого проекта в компании есть 2 обязательных специалиста, сотрудник, отвечающий за рекламное продвижение, а также менеджер, отвечающий за обсуждение и

согласование проекта, он должен понять, что в итоге хочет получить заказчик.

Состав работников проекта, создания и продвижения веб-сайта:

1. руководитель проекта;
2. аналитик;
3. проектировщик;
4. верстальщик;
5. программист;
6. веб-дизайнер;
7. тестер;
8. сео-специалист;
9. копирайтер.

Одной немало важной целью является стратегическое партнерство с различными компаниями. Стратегическое партнёрство, представляет собой сотрудничество физических и юридических лиц с долгосрочной перспективой.

Таковыми компаниями являются:

1. ООО "МЕХЭНЕРГО";
2. ООО "Северный поток";
3. ООО "МетроДеталь".

Компания на рынке занимает уверенную позицию, её роль при этом не просто оказать услугу, но и предоставить дополнительный сервис клиенту, для удержания источника постоянной прибыли. При разработке концепции работы компании, использовался собственный опыт работы на рынке продвижения сайтов с 2018 года. За время своей деятельности рекламная компания «Rosont», выступала на рынке в качестве студии, работающей напрямую, над продвижением веб-сайтов и других площадок.

Предприятие пользуется широким спросом на рынке продвижения сайтов, имеет ряд конкурентных преимуществ перед другими аналогичными компаниями:

1. готовность работать с любыми рекламными бюджетами;
2. главная цель работы, успешность бизнеса клиента;
3. выгодные цены, высокое качество в любой сфере;
4. надежное партнерство;
5. выполнение всех заказов в строго обговоренный срок;
6. цена ниже среднего на рынке;
7. бесплатная консультация по проекту;
8. бесплатный логотип в подарок при заказе веб-сайта;
9. большие скидки по разным услугам, для постоянных клиентов.

На текущий момент компания использует такие информационные системы как, MySQL база данных, а также активно использует программы офисного пакета ПО MSoffice: MSWord, MSeXcel и ПО Adobe: Photoshop, Illustrator.

При этом в качестве основных информационных каналов для отправки получения информации существуют:

1. телефон;
2. электронная почта;
3. группы в социальных сетях;
4. офис.

Недостатками, присущими объекту исследования, являются:

1. невозможность, быстрого поиска информации, из-за слишкомбольшого объёма информации;
2. низкая оперативность, приема заказов, снижающая количество обрабатываемых заказов;
3. неудобство, проведения онлайн-брифинга, по предлагаемой компанией услуге;
4. несовершенство, организации и технологий сбора, регистрации, хранения, обработки, выдачи и отображения информации.

1.2 Формулировка задачи

Все задачи можно разделить на 3 пункта.

Разработка макета веб-сайта для компании:

1. разработать прототип веб-дизайна;
2. разработать в соответствии с заказчиком веб-дизайнсайта;
3. спроектировать структуру сайта, проработать расположение элементов (кнопок, текстовых и графических блоков);
4. разработать навигацию сайта;
5. определить набор функциональных возможностей сайта;
6. заполнить страницы на сайте необходимым контентом;
7. реализовать возможность отправки онлайн-форм;
8. разработать планшетную, мобильную версию сайта.

Сайт должен обладать следующими качествами:

1. гибкостью, удобной для просмотра различной информации;
2. удобным расположением элементов для пользователя;
3. удобной и простой навигацией;
4. понятной и достоверной информацией;
5. быстрой загрузкой всех веб-страниц и прочих вкладок;
6. предоставление всех услуг с кратким описанием;
7. привлекательным дизайном, с использованием анимацией.

Целью создание корпоративного сайта является:

1. рассказать организации о себе (о предоставляемых услугах и товарах, о своем местоположении и т.д.);
2. заинтересовать клиентов, увеличить спрос на услуги;
3. увеличить узнаваемость компании, осуществить быстрый доступ к тому перечню товаров и услуг, которые она предлагает;
4. развить клиентскую базу, а также привлечь новых партнеров и сотрудников.

Эти задачи являются основными для рекламного агентства «Rocont» и планируется, что данная автоматизация принесет дополнительную выгоду от внедрения информационных технологий в управление организацией. Для большего привлечения клиентов, необходимо сделать веб-сайт оптимизированным под все устройства, чтобы информация хорошо отображалась, как на больших мониторах, так и на маленьких мобильных экранах.

Первой из важных целей является разработка привлекательного, продающего веб-дизайна и его удобного интерфейса, что поможет выделить организацию из числа подобных. Для этого дизайнер прорисовывают каждую страницу, размещают составные элементы в соответствии с утвержденным проектом. В итоге получается визуальное отображение сайта. Второй задачей будет верстка веб-сайта по сделанному веб-дизайну, для отображения страницы в браузерах. Специалист верстки разбивает текст на отдельные страницы, компоует его с иллюстрациями, прописывает структуру сайта, затем стилизует её по веб-дизайну.

1.3 Современные технологии, используемые для разработки веб-сайта

1.3.1 Методы разработки веб-сайтов

Веб-сайт — это совокупность программных, информационных, а также медийных средств, логически связанных между собой, по сути, это отражение успешности компании, её лицо.

Доменное имя, символическое имя, служащее для идентификации, областей, которые являются единицами административной автономии в сети интернет. Например, «yandex.ru» является доменным именем, которое объединяет все странички, находящиеся на сервере.

Веб-сервером называют, как программное обеспечение, выполнявшие функции сервера, так и домашний компьютер на котором установлено программное обеспечение, может быть использован в роли сервера.

Браузер специальная программа, предназначенная для просмотра веб-сайтов. Происходит это с помощью http запросов к серверу и получения от него данных, которые обрабатываются по специальным утвержденным стандартам и таким образом формируется веб-страница.

Скелет веб-сайта, который мы видим в браузере написан на языке HTML. HTML (язык гипертекстовой разметки), язык является стандартом разметки документов во всемирной паутине. Основное предназначение языка в разметке текста, для этого используются теги визуально они отображаются в угловатых скобках. HTMLдокументы будут корректно отображаться во всех браузерах и на мобильных устройствах. Все документы HTMLсодержат в себе стандартные элементы такие как титульная надпись, объявление кодировки, различные мета теги.

Основным способом верстки сайта является набор кода в текстовом редакторе. Это трудоёмкий и рутинный процесс подразумевает грамотное семантическое прописывание тегов, классов, продумывание иерархии сайта, умение работать стилями в CSS, работа с динамической анимацией по якорям, на языке программирования JavaScript.

Вторым способом верстки сайта является использование CMS. Это движок, который предоставляет возможности полуавтоматического создания веб-сайта по шаблону, создание сайта на CMS. Такой движок требует меньше знаний и времени, а также в CMS есть панель администратора с помощью, которой можно добавлять, удалять информацию, внедрять готовые плагины, скрипты в верстку веб-сайта.

Третьим способом является, возможность создавать веб-сайты на специальных сервисах, конструкторах, в них ты в визуальном редакторе подстраиваешь блоки под свой дизайн. В таких сервисах обычно нужны минимальные понимания верстки веб-сайта и работы с файлами. Этот метод позволяет достаточно быстро получить верстку со сложным оформлением, анимацией и скриптами.

Обозначим преимущества и недостатки веб-сайтов написанный на CMSи конструкторах. Что касается, достоинства CMSили сервиса конструктора, самым главным достоинством это будет скорость разработки, ручная верстка в редакторе займет гораздо больше времени. Главный недостаток таких методов — это качество сайта, если в CMSеще более читабельный и оптимизированный код, то в конструкторах в большинстве случаев код получается ужасный, также создание такого сайта будет стоить дороже.

Обозначим преимущества и недостатки веб-сайтов, написанных вручную. Созданный таким образом веб-сайт гораздо сложнее взломать, так как нет MySQLбазы данных и отсутствует файл конфигурации, а именно с помощью этих файлов в первую очередь хакеры пытаются взломать веб-сайт.

Веб-сайты, сверстанные вручную отображаются гораздо быстрее если их сравнивать с CMSсистемами, сервисами конструкторами. Так получается потому что, код получается более оптимизированным, файлы намного компактнее и не надо делать запросы MySQLбазу данных.

Скорость отображение веб-сайта положительно влияет на будущего клиента если загрузка долгая, то это негативный фактор, скорость влияет на позицию выдачи при поисковых запросах. При одинаковых условиях веб-сайт, написанный на вручнуюбудет выше. Веб-сайт занимает меньше места что позволяет экономить на тарифах хостинга.

Для описание внешнего вида элементов страниц применяется CSS (каскадные таблицы стилей).CSSдовольно просто использовать в HTMLдокументах. Для этого нужно прописать в head(предназначен для хранения других элементов), тег style(применяется для определения стилей элементов веб-страницы) и внутри этого тега прописывать стили.

CSS довольно простой формальный язык, который был создан для описания внешнего вида документов. Язык прост и состоит из самобытных примитивных конструкций, которые не так сложны в запоминание,

выполняющие определенные задачи. Самое сложное в запоминании не написание конструкций. Например, чтобы задать цвет тексту, достаточно, сделать перевод на английский в результате получится color и название цвета.

CSSселектор (от слова выбирать) – это конструкция с которой начинается каждый блок объявлений, служит для выборки элемента или однотипных элементов на странице, для дальнейшей стилизации. Чаще всего в качестве селектора используется определенный класс тега. Пример определения класса тегом приведен на рисунке 1.

```
1 <!DOCTYPE html>
2 <html lang="ru">
3 <head>
4   <meta charset="UTF-8">
5   <title>Цвет текста</title>
6 </head>
7 <body>
8 <style>
9
10  .description {
11    color: cadetblue;
12  }
13
14 </style>
15
16 <p class="description">Lorem ipsum dolor sit amet, consectetur adipiscing elit. Magnam eius neque perspiciatis
17 laborum dolor tempore sit iste accusantium voluptatibus nam ipsum accusamus eos est distinctio enim, sint
18 repellendus dicta praesentium! Officiis, obcaecati, excepturi! Beatae veritatis distinctio id obcaecati, pariatur
19 odit modi, at, magni voluptate, possimus omnis culpa? Porro, explicabo assumenda.</p>
20 <p class="description">Lorem ipsum dolor sit amet, consectetur adipiscing elit. Magnam eius neque perspiciatis
21 laborum dolor tempore sit iste accusantium voluptatibus nam ipsum accusamus eos est distinctio enim, sint
22 repellendus dicta praesentium! Officiis, obcaecati, excepturi! Beatae veritatis distinctio id obcaecati, pariatur
23 odit modi, at, magni voluptate, possimus omnis culpa? Porro, explicabo assumenda.</p>
24 <p class="description">Lorem ipsum dolor sit amet, consectetur adipiscing elit. Magnam eius neque perspiciatis
25 laborum dolor tempore sit iste accusantium voluptatibus nam ipsum accusamus eos est distinctio enim, sint
26 repellendus dicta praesentium! Officiis, obcaecati, excepturi! Beatae veritatis distinctio id obcaecati, pariatur
27 odit modi, at, magni voluptate, possimus omnis culpa? Porro, explicabo assumenda.</p>
28 </body>
29 </html>
```

Рисунок 1 – Определенный класса тега

Здесь селектором будет класс descriptionтега p, который получает необходимое оформление вHTMLдокументе, в теге <style></style>. В данном случае цвет текста – cadetblue. Соответственно, если на странице есть несколько тегов (не только p)с классом description, все элементы получат одинаковое оформление,цвет текста серо-голубой.

В проектах где, большой объём работы, использование CSSне целесообразно потому что, получатся большое и не структурированное количество кода, в котором сложно ориентироваться. Для решения этой проблемы есть CSS препроцессор. CSS препроцессор — это надстройка над CSS, которая добавляет ранее недоступные возможности для CSS, с

помощью новых синтаксических конструкций. CSS препроцессоры преобразуют код, написанный с использованием препроцессорного языка, в чистый и валидный CSS-код.

При помощи препроцессоров вы можете писать код нацеленный на:

1. читабельность для человека;
2. структурированность и логичность;
3. производительность.

Самым популярным предлагаемым функционалом любого CSS-препроцессора является, возможность вкладывать селекторы друг в друга. Оптимальным CSS-препроцессором для маленьких и средних проектов будет LESS, его функционала должно хватать.

В CSS-препроцессоре LESS, есть примеси, если говорить совсем кратко, то используя примеси (от англ. Mixins), можно сделать код повторно используемым. Это помогает избежать вспомогательных классов в разметке или дублирования свойств от селектора к селектору.

Переменные в LESS, переменные нужны для организации и упрощения написания кода и позволяют предварительно задать популярные значения, а затем повторно использовать их в любом месте кода, просто написав имя переменной. Это удобно, потому что нам при необходимости не придётся отыскивать и править множество значений, достаточно изменить его в одном месте и значения везде заменятся автоматически.

LESS также позволяет нам выполнять такие операции как сложение, вычитание, умножение и деление числовых значений, цветов и переменных в таблице стилей. Допустим, мы объявили переменную `padding` которая применяется ко всем нашим элементам `H1` (заголовок первого уровня). Но теперь мы хотим добавить дополнительный отступ для нашего заголовка на главной странице.

Для того, что ускорить разработку, необходимо использовать БЭМ (блок, элемент, модификатор). БЭМ — это не только простое именование

классов, это компонентный подход к работе. Компонентный подход — это когда мы делим разрабатываемый интерфейс на отдельные блоки. При надобности данный блок можем использовать в другой части сайта или приложения без дублирования стилей. Сейчас, конечно же, мало, что понятно, но мы будем разбирать примеры.

Основу БЭМ методологии составляют 3 составляющие:

1. блок;
2. элемент;
3. модификатор.

Блоком — это функционально независимый компонент, который может быть повторно использован на любой странице сайта и в любой части документа. Блок именуется только атрибутом `class`. Название блока (а именно присваиваемый класс) должно характеризовать смысл или предназначение блока.

Элемент — это составляющая блока, которая не может быть использована в отрыве от него. Именование элемента выглядит следующим образом: имя-блока__имя-элемента. То есть название элемента отделяется от имени блока двойным нижним подчеркиванием (`__`).

Модификатор — это БЭМ сущность, которая определяет вид, состояние или поведение блока, или элемента. Модификатор отделяется от названия блока или элемента одинарным нижним подчеркиванием (`_`). Структура имени модификатора будет следующим, для блока: имя-блока_модификатор.

Для ускорения верстки использования удобной сетки можно использовать библиотеку. Bootstrap— свободный набор инструментов, для создания веб-сайтов и веб-приложений. Включает в себя HTML и CSS-шаблоны оформления для типографики, веб-форм, кнопок, меток, блоков навигации и прочих компонентов веб-интерфейса, включая JavaScript-расширения.

Преимущества библиотеки bootstrap:

1. высокая скорость разработки макетов страниц сайта, bootstrap содержит огромный набор готовых решений и элементов;
2. кроссбраузерность и адаптивность сайта, все элементы библиотеки адаптивны под все устройства и корректно отображаются во всех современных браузерах;
3. легкость в использовании, даже человек, имеющий базовые знания о HTML и CSS, может свободно создавать веб-страницы с использованием библиотеки;
4. простота в обучении, у bootstrap очень хорошая документация с большим количеством примеров готового кода.

Для динамических элементов в проекте понадобится JavaScript. JavaScript— язык программирования, с помощью которого создаются всевозможные приложения и красочные анимации, они могут либо встраиваться в html-код страницы, либо вызываться с помощью простых команд как отдельный программный файл. Использование технологии JavaScript при верстке сайта позволяет создавать различные формы обратной связи, регистрационные анкеты, подсказки и предупреждения, вносить изменения в стили элементов страниц, удалять или добавлять теги, манипулировать элементами html-кода, записывать данные о посетителях и многое другое.

JavaScript поддерживается всеми браузерами и интегрируется с ними. Однако возможности сценариев JavaScript не безграничны. Так, например, JS не выполняет кроссдоменные запросы, ему недоступны файлы, кроме cookie, расположенные не на компьютере пользователя, не защищает страницу от копирования текста.

С помощью него доступны к исполнению следующие функции:

1. возможность изменять страницы браузеров;
2. добавление или удаление тегов;
3. изменение стилей страницы;
4. информация о действиях пользователя на странице;

5. запрос доступа к случайной части исходного кода страницы;
6. внесение изменений в этот код;
7. выполнение действия с cookie-файлами.

1.3.2 Проектирование веб-дизайна

Одной из важнейших задач при разработке веб-дизайна ресурса является, реализация такой его функции как коммуникация. Иначе говоря, веб-дизайн-сайта должен быть таков, чтобы оказавшийся здесь по новому запросу пользователь, не только не покинул его, но и захотел бы более подробно ознакомиться с содержанием ресурса и прочей предлагаемой веб-сайтом информации. Веб-сайт не может долго и успешно функционировать, если на каждом этапе разработке, его дизайн не будет гармонировать с его содержанием. Разработка дизайна сайта — это создание прямого и ровного пути между посетителем сайта и целью, которой является искомые данные. Поэтому очень важно при проектировании веб-сайта хорошо понимать на какую аудиторию, он рассчитан какие потребности возникают у пользователей. Все эти задачи в веб-дизайне решает Ux/UI.

UX – это UserExperience (дословно «опыт пользователя»). То есть это то, какой опыт/впечатление получает пользователь от работы с вашим интерфейсом. Удастся ли ему достичь цели и на сколько просто или сложно это сделать.

UI это UserInterface (дословно «пользовательский интерфейс») — то, как выглядит интерфейс и то, какие физические характеристики приобретает. Определяет, какого цвета будет ваше «изделие», удобно ли будет человеку попадать пальцем на кнопки, читабельным ли будет текст и тому подобное.

UXэто понятие, которое применимо не только в веб-дизайне, но и в других областях. Опыт взаимодействия – это когда специалист на заводе управляет сложной техникой посредством нажатия пары кнопок, причем кнопка отключения имеет красный цвет и значительней крупнее. UX – это когда вы срезаете путь до дома тропинкой, хотя есть красивый и чистый

тротуар, когда вы ставите стаканчик с кофе в специальные углубления на панели задач в салоне вашего авто, это когда педаль тормоза шире педали газа. Все перечисленное это опыт взаимодействия пользователя. Примеров может быть масса, но суть одна: UX – это способ достижения цели пользователем, максимально комфортным способом. И при создании веб-дизайна, надо продумывать пользовательское удобство. В результате работы над UXмы должны получить рабочий прототип. Так как речь идет о веб-дизайне, таким прототипом является готовый макет.

UXв веб-дизайне – это фундамент, на котором строится дизайн веб-сайта или приложения. Это проектирование интерфейса с учетом потребностей людей, для которых вы создаете дизайн веб-сайта. Кто конечный пользователь, какие цели человек преследует, используя ваш интерфейс, как помочь ему максимально быстро и комфортно получить результат? Вот основные задачи, которые решает UX.

Прямая обязанность UX/UI дизайнера — это, например, продать товар или услугу через интерфейс. Именно на основе работы UX/UI дизайнера пользователь принимает решение: «Быть или не быть?» Нравится или не нравится. Купить или не купить.

Главная задача веб-дизайнера, максимально повысить уровень удовлетворенности конечного пользователя, от взаимодействия с продуктом. Под продуктом мы будем понимать любой объект взаимодействия с пользователем, будь то реальный продукт услуга веб-приложении или сайт.

Специфическая задача веб-дизайнера заключается в том, чтобы уметь разработать стилевое оформление проекта с учетом специфики интернета. То есть, кроме того, чтобы оформление проекта было просто стильным, оно должно соответствовать стандартным требованиям, предъявляемым сетью. Графические элементы сайта при этом должны быть правильно оптимизированы: правильно определён цвет и шрифт. Также необходимо учитывать тот факт, что пользователь увидит графическое воплощение

проекта на мониторе, а не на бумаге, а также будет непременно взаимодействовать с ним.

В начале своего исследования нужно пообщаться с заказчиком и обсудить детали работы бизнеса. Он является специалистом в своем продукте и у него есть опыт работы с клиентами, он знает свою аудиторию. Необходимо записать все ключевые моменты для дальнейшего анализа, также нужно изучить опыт аналогичных продуктов.

После того как вы определили аудиторию, определили цели и ценности персон можно приступить к созданию прототипа сайта так называемому вайрфреймингу. Прототип нужен, для того чтобы сфокусироваться на смысле и продумать интерфейс на концептуальном уровне, прежде чем переходить к графическому оформлению. Хорошо составленный прототип является полноценным каркасом сайта, на который потом легко «надевается» веб-дизайн.

Для создания каркаса можно воспользоваться специальным для этого сервисом или нарисовать от руки на бумаге в клеточку после чего отсканировать и положить в папку с материалами проекта. Отлично подходит онлайн-сервис wireframe.cc так как можно в реальном времени вносить правки и согласовываться клиентом, также прототип можно онлайн, передать клиенту для заполнения текстами. Нужно проработать все страницы веб-сайта или приложения и только после того как каркас всего веб-сайта готов, переходить к следующему шагу.

После того как был проработан UX, сделан прототип блоков и написаны все текста можно переходить к UI. Визуализация — это прорисовка каркасов создание единого стиля, оформление контента. Другими словами, начинаем работу над UI. Чаще всего для визуализации используется Adobe Photoshop, Sketch.app, но лучше использовать редакторы, которые предназначены под текущую задачу, для этого подойдет Figma.

В веб-дизайне есть правила хорошего тона. Разрабатывая дизайн веб-сайта, веб-дизайнер придерживается определенных негласных правил при

оформлении. Межстрочный интервал отступы шрифты размеры элементов это все поддается измерению и есть откровенно плохие реализации. Поэтому можно составить свод правил, придерживаясь которых поможет не сделать плохую визуализацию вашего дизайна.

1.4 Этапы создания веб-сайта

1. Создание прототипа веб-дизайна

Прототип веб-дизайна — это как правило образец, наброски расположения блоков будущего веб-дизайна, рисуется он в графических редакторах таких как Wireframe.cc, Marvel, Moqups. Создание прототипа является одним из главных этапов тестирования веб-дизайна. На этом этапе формируется внешний вид и добавляется все графика, уточняются все моменты и детали.

2. Создание веб-дизайна

На этом этапе происходит проектирование логической структуры веб-страниц по готовому прототипу, придумывание наиболее удобных решений для подачи информации в веб-дизайне, полное художественное оформление проекта. Рисуется веб-дизайн в графических редакторах таких как AdobePhotoshop, Sketch или более подходящим для этой задачи редактором, проектирования интересов Figma.

3. Верстка проекта

Макет веб-сайта, нарисованный в графическом редакторе нарезается на графические элементы, описывается структура сайта на языке HTML, далее по готовности скелета, описываются нужные стили для каждого блока на языке CSS, также для интерактивности макета при верстке могут быть использованы такие технологии, как javascript, если необходима некая динамическая погрузка или сложная анимация.

4. SEO оптимизация верстки

SEO (действия, направленные на повышение позиций сайта в выдаче поисковых систем), что ведет к увеличению посещаемости с поисковиков. К

SEO можно также отнести работы над сайтом связанные с повышением таких показателей, тематический индекс цитирования Яндекса, PageRank, Google, траст сайта. Работа внутри веб-сайта, то есть внутренняя оптимизация, исправление ошибок, допущенных при разработке сайта, наполнение новым контентом, изменение HTMLкода страниц, перелинковка. Кроме того, необходимо учитывать специфику поисковых алгоритмов, SEOоптимизация веб-сайтов под Яндекс и под Googleотличается некоторыми нюансами.

5. Привязка домена и размещение веб-сайта на хостинге.

После приобретения домена и хостинга, нужно провести окончательный этап настройки работы сайта. При этом учитывая, что хостинг и домен могут быть оформлены у разных интернет регистраторов, панели управления могут сильно визуально отличаться.

Настройка работы сайта состоит из двух этапов:

1. добавить домен на сервер хостинга;
2. указать в домене IP адрес сервера хостинга.

1.5 Программное обеспечение, необходимое для создания веб-сайта

1.5.1 Текстовые редакторы

Программы, которые необходимы для написания и редактирования текстовых документов. Для этой задачи подойдет даже самый простой «блокнот», но для удобства лучше воспользоваться специальными текстовыми редакторами, которые предназначены для написания HTMLкода. Такие программы подсвечивают теги, помогают находить ошибки в коде и лучше структурировать код. Также в них есть множество полезных функций, позволяющих ускорить процесс написания кода.

Примеры таких программ:

1. notepad++;
2. aptana Studio;

3. sublime text;

4. brakets.

Для наших целей отлично подойдет Brackets, так как в нем можно быстро установить необходимые для работы плагины, также очень гибкая настройка и множество полезных функций позволяющий ускорить процесс верстки.

1.5.2 Графические редакторы

При создании прототипа и макета веб-сайта можно воспользоваться графическими редакторами это программы, позволяющие работать с графикой (создавать, редактировать различные изображения). Так же, как и текстовые редакторы, они бывают с различными функционалами и разным применением. К простому редактору в первую очередь относятся WindowsPaints. К сложным графическим редакторам, которые работают с растровой графикой относятся AdobePhotoshop и Gimp. Для векторной графики используется AbodeIllustrator, Figma, CorelDraw. Также можно использовать онлайн-сервис wireframe.cc для создания прототипа веб-сайта. В нем можно набросать структуру веб-сайта, обозначить расположение ключевых элементов их размеры, примерные цвета.

1.5.3 Локальные сервера

Локальный сервер — это набор программ, которые создают аналог хостинга непосредственно на вашем компьютере и позволяют создавать и тестировать динамические сайты с использованием серверных языков программирования и базы данных.

Локальный сервер OpenServerPanel — это портативная серверная платформа и программная среда, созданная специально для веб-разработчиков с учётом их рекомендаций и пожеланий. Программный комплекс имеет богатый набор серверного программного обеспечения, удобный, многофункциональный продуманный интерфейс, обладает мощными возможностями по администрированию и настройке компонентов.

Платформа широко используется с целью разработки, отладки и тестирования веб-проектов, а также для предоставления веб-сервисов в локальных сетях.

Локальный сервер AMPPS — это сборник приложений, которые создают целую библиотеку для работы веб-мастера. Каждое приложение можно установить, протестировать и использовать в своих целях, причем делать это очень быстро. Данная сборка имеет свои плюсы и преимущества перед другими. Стоит отметить поддержку распространенных CMS, веб-форумов, блог-платформ, фотогалерей, досок объявлений и другое. Количество доступных приложений насчитывает более 200 штук.

Локальный сервер Denwer — это комфортные условия при удаленной работе сразу над несколькими независимыми проектами и возможность размещения на флеш-накопителе. Кроме того, эмулятор веб-сервера Denwer может быть использован для создания собственного хостинга, а также при создании баз данных для веб-сайтов, используя при этом язык SQL.

2. Разработка веб-сайта

2.1 Разработка прототипа веб-сайта

После общения с владельцем компании, в соответствии с требованиями были выявлены ключевые моменты, которые должны быть отражены при разработке веб-сайта. Так как веб-сайт служит для привлечения новых клиентов, поэтому сайт должен быть удобен в поисках необходимой информации. Информация, имеет ключевые моменты, логотип, наименование и описание компании, клиенты с которыми работали, преимущества перед другими рекламными агентствами, описание услуг и их стоимость. На главной странице веб-сайта должна быть удобная навигационная панель, краткое описание чем занимается компания, контактные данные, форма обратной связи. В нижней части должна быть полная навигационная карта веб-сайта, ссылки на социальные сети, адрес, контактные данные, политика конфиденциальности.

Когда мы определились с тем, что должно быть на странице, можно приступать к разработке прототипа веб-сайта. Разработка прототипа делается только основных страниц навигации. Для начала определимся с общей структурой веб-сайта. Это нужно чтобы изначально понять, каким должен быть объём будущего веб-сайта, какие технологии будут использоваться и какие функции будет включать веб-сайт. Существует несколько основных структур веб-сайтов.

Линейная структура — это когда каждая страница ссылается на другую страницу и на главную. Такую структуру хорошо применять на сайтах-презентациях, портфолио и других специфических продуктах, которые преследуют цель ознакомить посетителя со всеми страницами в определенной последовательности. Вес страниц здесь перетекает от главной к последней странице через все остальные. Успешно продвигать можно только главную, поэтому для привлечения посетителей эта структура не очень подходит. Оставим ее тем, кто использует сайты для своих целей и не

собирается получить трафик из поиска. Пример линейной структуры веб-сайта приведен на рисунке 2.

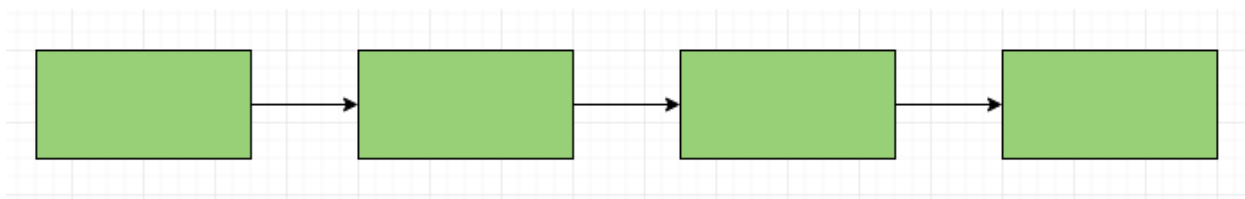


Рисунок 2 – Линейная структура веб-сайта

Разновидностью выше рассмотренной структуры веб-сайта является, линейная структура ресурса с ответвлениями. Она подобна дороге с большим количеством ответвлений от нее. Пользователь перемещается с одной страницы на другую, в строго определенном порядке. Однако при необходимости, он всегда сможет перейти на другую ветвь и без усилий вернуться назад.

Главным достоинством линейной структуры с ответвлениями являются, относительно несложная возможность веб-мастерам создать ее на основе обычной линейной структуры. По мере раскрутки веб-сайта в этом довольно часто возникает потребность. Контент сильно разрастается и возникает вопрос улучшения навигации. Поэтому важен такой процесс, как нарисовать структуру веб-сайта. Пример линейной структуры с ответвлениями веб-сайта приведен на рисунке 3.

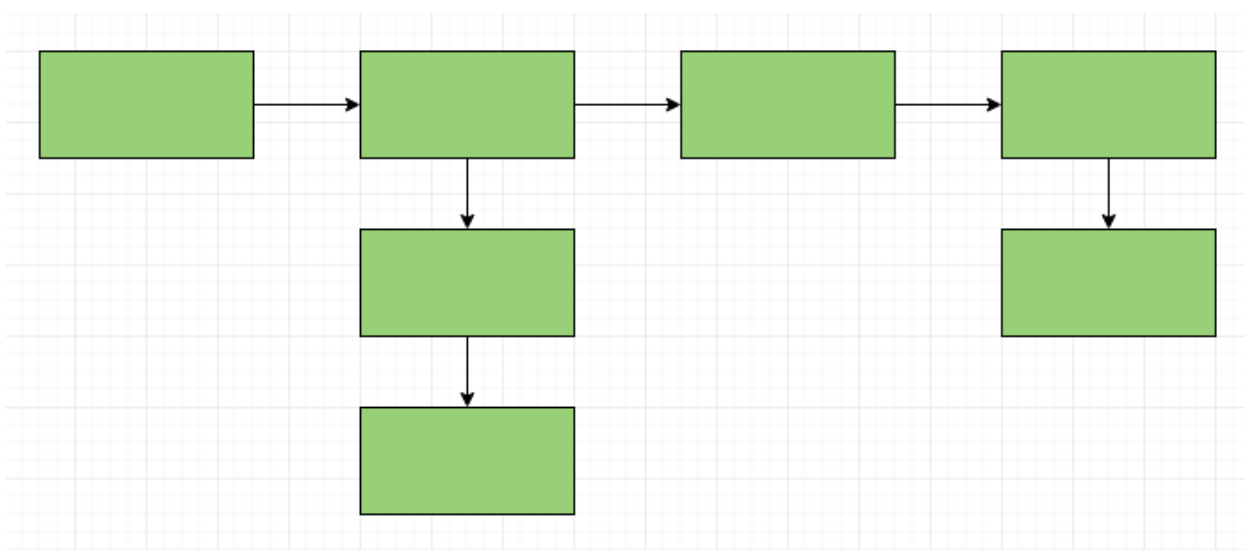


Рисунок 3 – Линейная структура с ответвлениями веб-сайта

Древовидная структура — это древовидная структура является наиболее универсальным вариантом и именно ее вы будете применять в 99% случаев. Смысл в том, что для каждого направления у вас будет своя ветка, для каждой услуги или товара у вас будет отдельное ответвление. То есть, те самые привычные нам разделы и подразделы. Эта структура позволяет передавать дополнительный вес как на главную, так и на разделы (каждая страница раздела будет ссылаться не только на главную, но и на свой раздел, достаточно настроить хлебные крошки. Пример древовидной структуры веб-сайта приведен на рисунке 4.

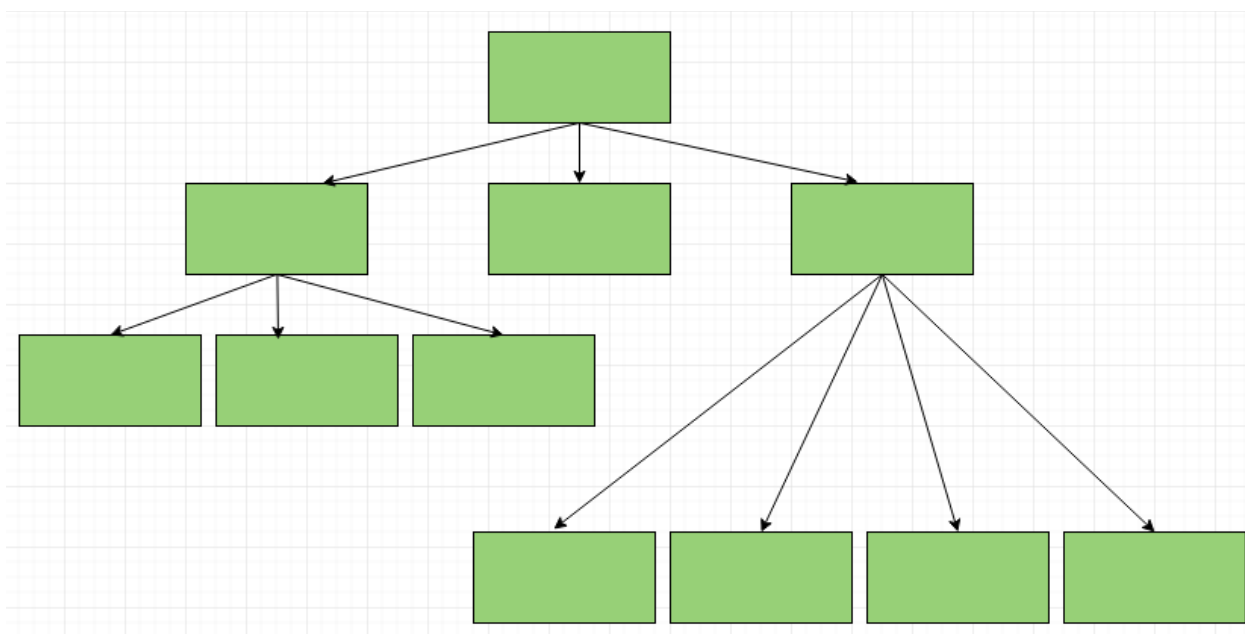


Рисунок 4 – Древовидная структура с ответвлениями веб-сайта

Блочная структура — это когда все страницы ссылаются на несколько других, которые равнозначны между собой. Такую структуру неплохо применять для конкретного продукта, когда каждую страницу можно использовать, как описание какого-то отдельного свойства/достоинства и их совокупностей. С распределением веса здесь все вполне неплохо, страницы уже перелинкованы и отдают свой вес на главную, что позволяет продвигать ее более эффективно. Но такая структура весьма специфична и применять ее можно далеко не везде. Пример блочной структуры веб-сайта приведен на рисунке 5.

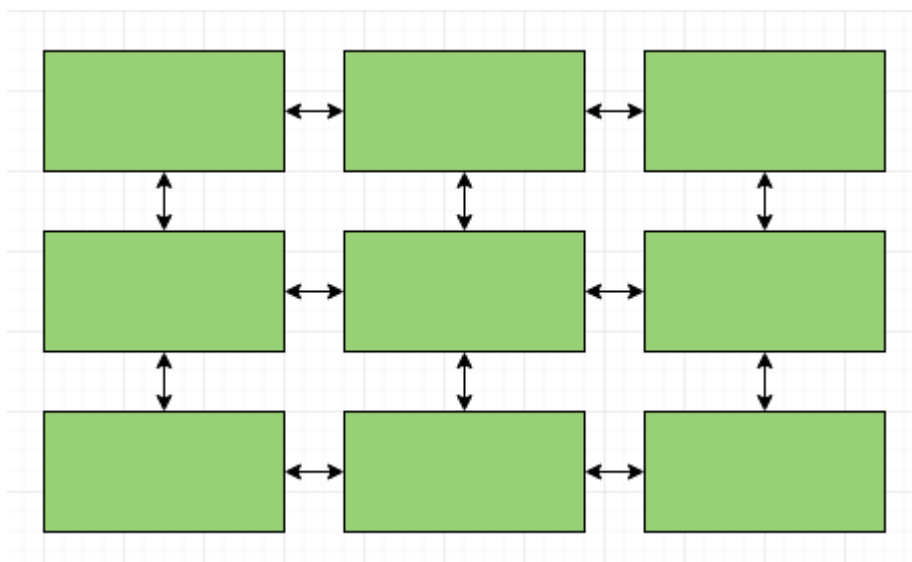


Рисунок 5 – Блочная структура веб-сайта

Структуру веб-сайта можно разработать самостоятельно, если использовать программу для создания структуры веб-сайта. Для нашего веб-сайта отлично подойдет древовидная структура, так как она подходит для небольшого корпоративного веб-сайта. Достоинством этой структуры в проекте будет удобство навигации, перемещения пользователя по веб-сайту, а это одна из главных задач.

Для веб-сайта рекламного агентства хорошим тоном будет использование современного оформления в веб-дизайне. По сути, графика, оформление ресурса и его внешний вид сами по себе не являются основным пунктом и стартовой точки при проектировании веб-сайта.

Сюда включаются:

1. графический интерфейс, который мы видим, открыв сайт;
2. проектирование навигационной системы сайта;
3. контент-проект ресурса, то есть информационное содержимое сайта, упитывающееся при его разработке.

Главная страница веб-сайта является, важнейшей страницей на веб-сайте. Именно после впечатления главной страницы потенциальный покупатель решит оставаться на веб-сайте или нет. Любая главная страница должна отвечать нескольким требованиям:

1. ясно отражать тематику веб-сайта;

2. быть понятной целевой аудитории не содержать ничего лишнего и отвлекающего;
3. содержать некое ценное предложение, которое сразу заинтересует потенциального клиента;
4. она должна быть удобна и понятна пользователю чтобы ему не приходилось долго искать элементы управления также желательно удобное управление на мобильных устройствах;
5. иметь однотипный веб-дизайн.

Остальные страницы веб-сайта должны быть объединены общим дизайном на них должны быть элементы меню ссылки на главную страницу и название ресурса. В остальных моментах страницы не должны дублироваться, то есть иметь уникальное содержание. Пример структуры страниц веб-сайта компании «Rocont» приведен на рисунке 6.

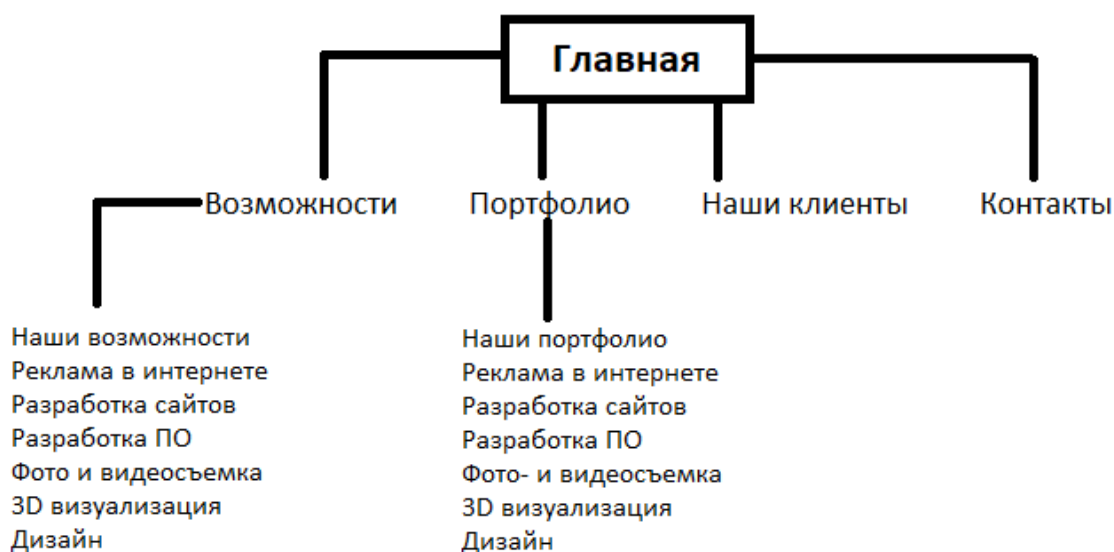


Рисунок 6 – Структура страниц веб-сайта компании «Rocont»

Веб-сайт должен поддерживаться на всех популярных сегодня операционных системах – Windows, Linux, Android, IOS. Также он должен корректно отображаться во всех популярных браузерах и на всех разрешениях экранов. Для того применяется адаптивная верстка, которая позволяет веб-сайту менять размеры и расположение своих элементов в зависимости от того на экране какого разрешения он отображается. К тому

же веб-сайты с использованием адаптивной верстки становятся поисковыми системами на более высокие позиции. После того как определены основные моменты внешнего вида веб-сайта можно приступить к ее прототипу. Разработка прототипа будет проводиться в онлайн-сервисе wireframe.cc.

Приступаем к прототипу главного экрана веб-сайта, на который будет попадать клиент при переходе на веб-сайт. Пример первого экрана прототипа главной страницы приведен на рисунке 7.

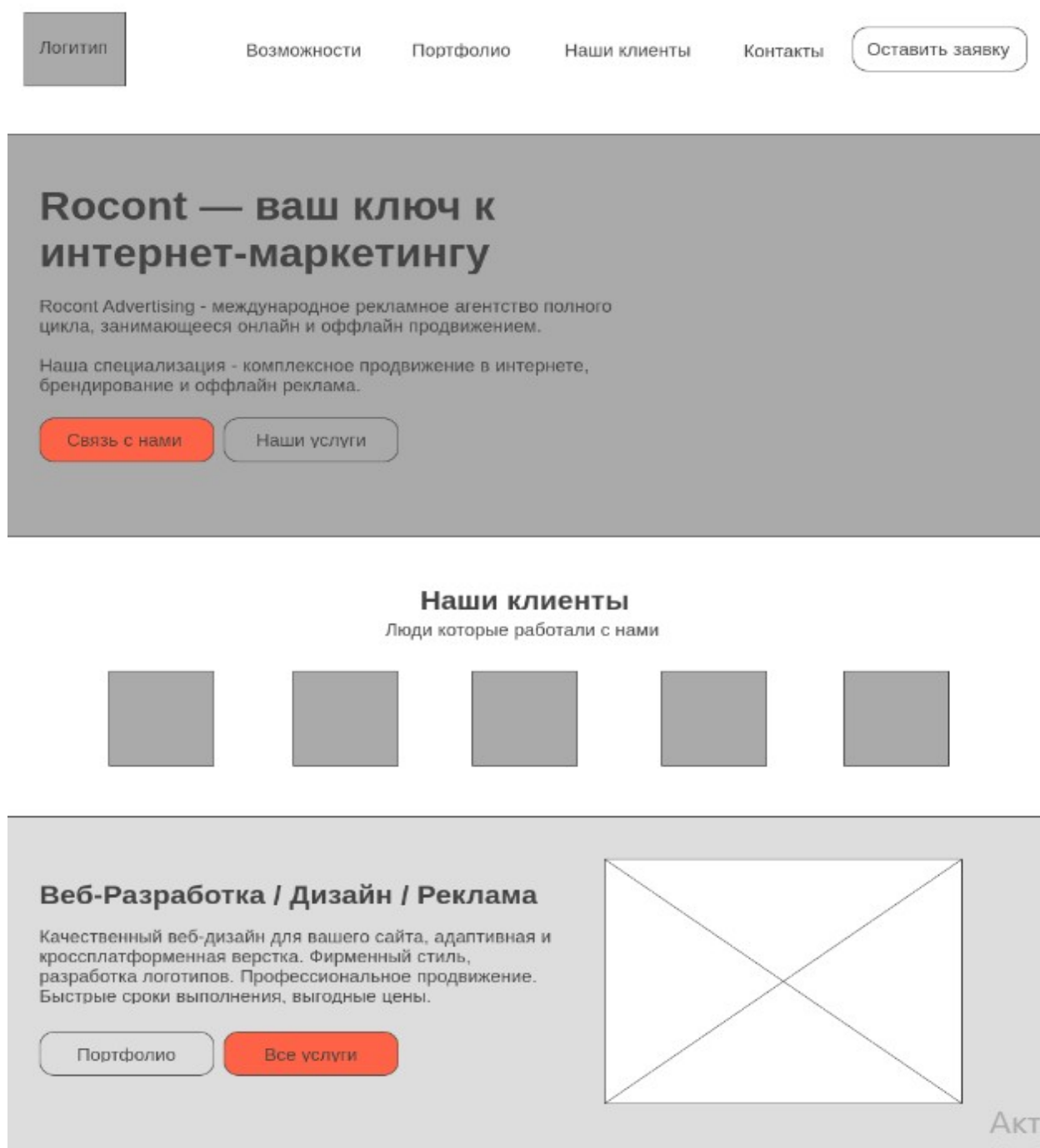


Рисунок 7.1 – Прототип главная страница, верхняя часть

В верхней части главной страницы представлен логотип, навигация, кнопка с всплывающим окном обратной связи, обозначен род деятельности и описание компании, блок с логотипами довольных клиентов. Последний блок описывает одну из популярных услуг сайта и дано краткое описание, кнопка с переходом в портфолио и кнопка с переходом во все услуги.

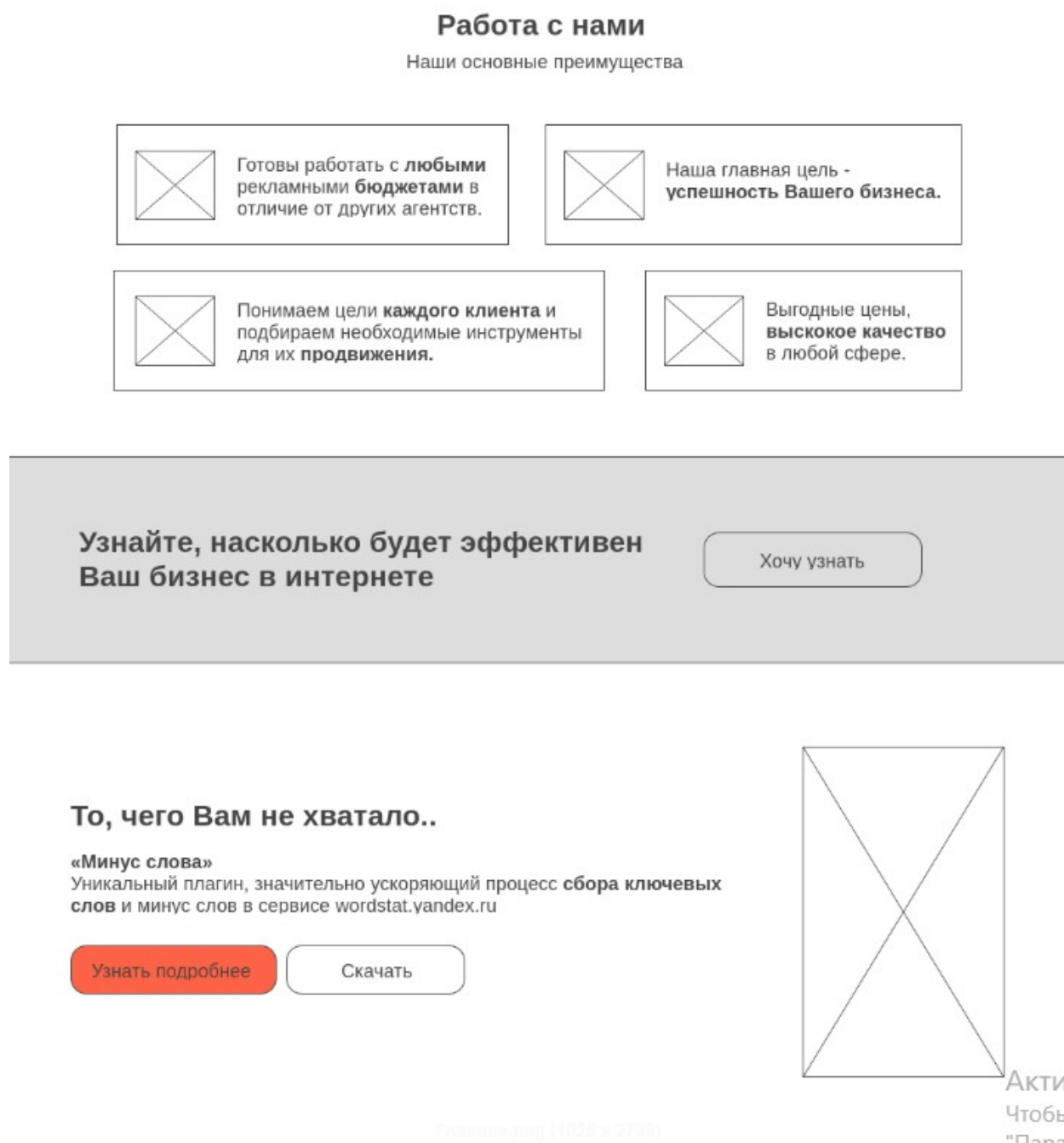


Рисунок 7.2 – Прототип главная страница, нижняя часть

В нижней части главной страницы, представлены преимущества работы с компанией, 4 ключевых преимущества, блок для привлечения внимания клиентов который направляет на подробную информацию. Также блок с специальным программным обеспечением по сбору ключевых слов.

Приступаем к прототипу раздела возможности, в котором находится возможности, которые предоставляет компания. Второй экран прототипа страницы возможности приведен на рисунке 8.

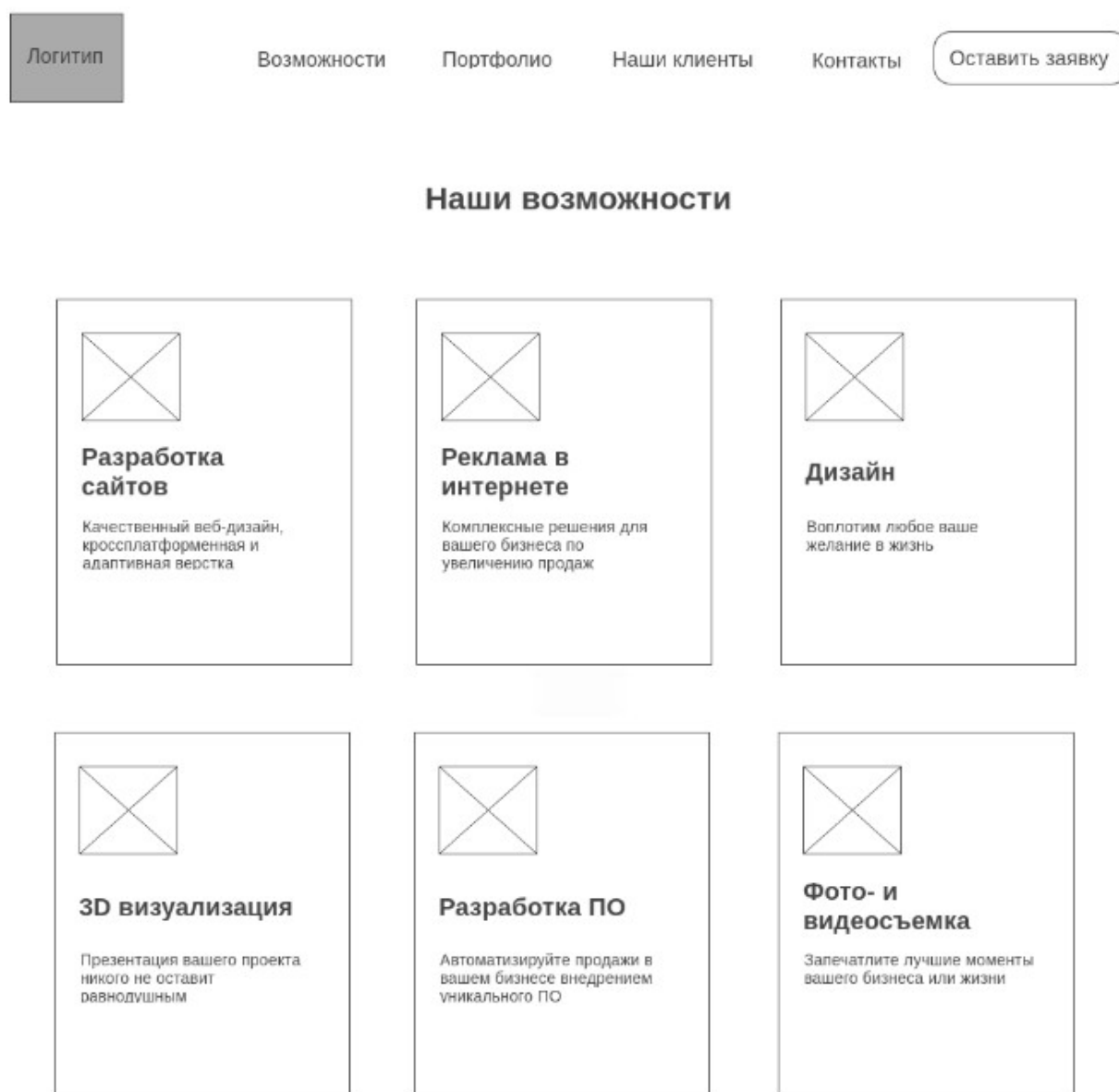


Рисунок 8 – Прототип страницы наши возможности

В разделе наши возможности, прописывается вся информация по описанию всех услуг, которые представляет компания. Блоки с

возможностями включают в себя картинку, заголовок и описание. В возможности компании входит, разработка веб-сайтов, реклама в интернете, дизайн, разработка программного обеспечения, фото и видеосъемка.

Приступаем к прототипу раздела портфолио, в котором находятся витринные работы компании. Третий экран прототипа страницы портфолио приведен на рисунке 9.



Рисунок 9 – Прототип страницы наше портфолио

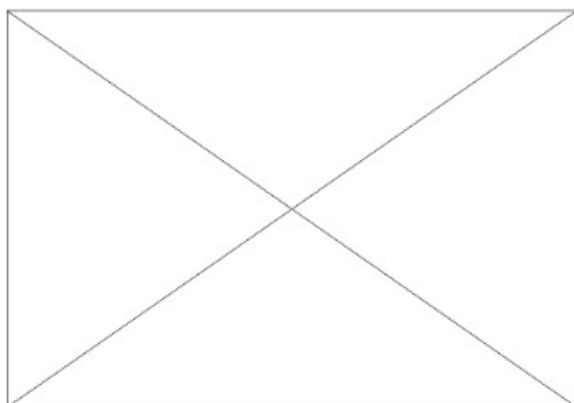
В разделе наше портфолио, в котором находиться, выполненные компанией работы. В портфолио входят такие работы как, реклама в интернете, разработка веб-сайтов, разработка программного обеспечения, фото и видеосъемка, дизайн.

Приступаем к прототипу раздела контакты, в котором написаны основные контактные данные компании. Четвертый экран прототипа страницы контакты приведен на рисунке 10.

Контакты

Адрес офиса в СПб:

Санкт-Петербург, БЦ Кантемировский



Информационная линия:

Городской номер: +7 (812) 629-6097
Мобильный номер: +7 (911) 211-78-10
info@rocont.ru



Обратная связь

Ответим на любой вопрос

Ваше Имя:

Ваш Email:

Ваш Телефон:

Ваш вопрос:

Рисунок 10 – Прототип страницы контакты

В разделе контакты находиться, форма обратной связи для ответов на вопросы клиентов, также в информации указан, адрес, городской номер телефона, мобильный номер телефона, почта.

2.2 Создание дизайна веб-сайта

Как мы видим веб-дизайн получается достаточно обильный по информации, хорошим расположением для удобного просмотра. Теперь, когда готов прототип можно проступить к более тщательной прорисовке и определению окончательного дизайна будущего веб-сайта. Разработка производилась в программе Figma, так как эта программа позволяет быстро и качественно выполнить её. В дизайне используем мягкие цвета. В качестве акцентного цвета используем цвет, взятый из логотипа компании.

Приступаем к веб-дизайну главного экрана веб-сайта. Веб-дизайн первого экрана приведен на рисунке 11.

ROCONT — ВАШ КЛЮЧ К ИНТЕРНЕТ-МАРКЕТИНГУ

Rocont Advertising - международное рекламное агентство полного цикла, занимающееся онлайн и офлайн продвижением.

Наша специализация - комплексное продвижение в интернете, брендинг и офлайн реклама.

[Связь с нами](#)[Наши услуги](#)

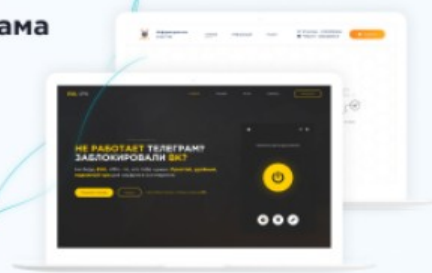
Наши клиенты

Люди которые работали с нами



Веб-Разработка / Дизайн / Реклама

Качественный веб-дизайн для вашего сайта, адаптивная и кроссплатформенная верстка. Фирменный стиль, разработка логотипов. Профессиональное продвижение. Быстрые сроки выполнения, выгодные цены.

[Портфолио](#)[Все услуги](#)

Активация
Чтобы активировать

Рисунок 11.1 – Веб-дизайн главная страница, верхняя часть

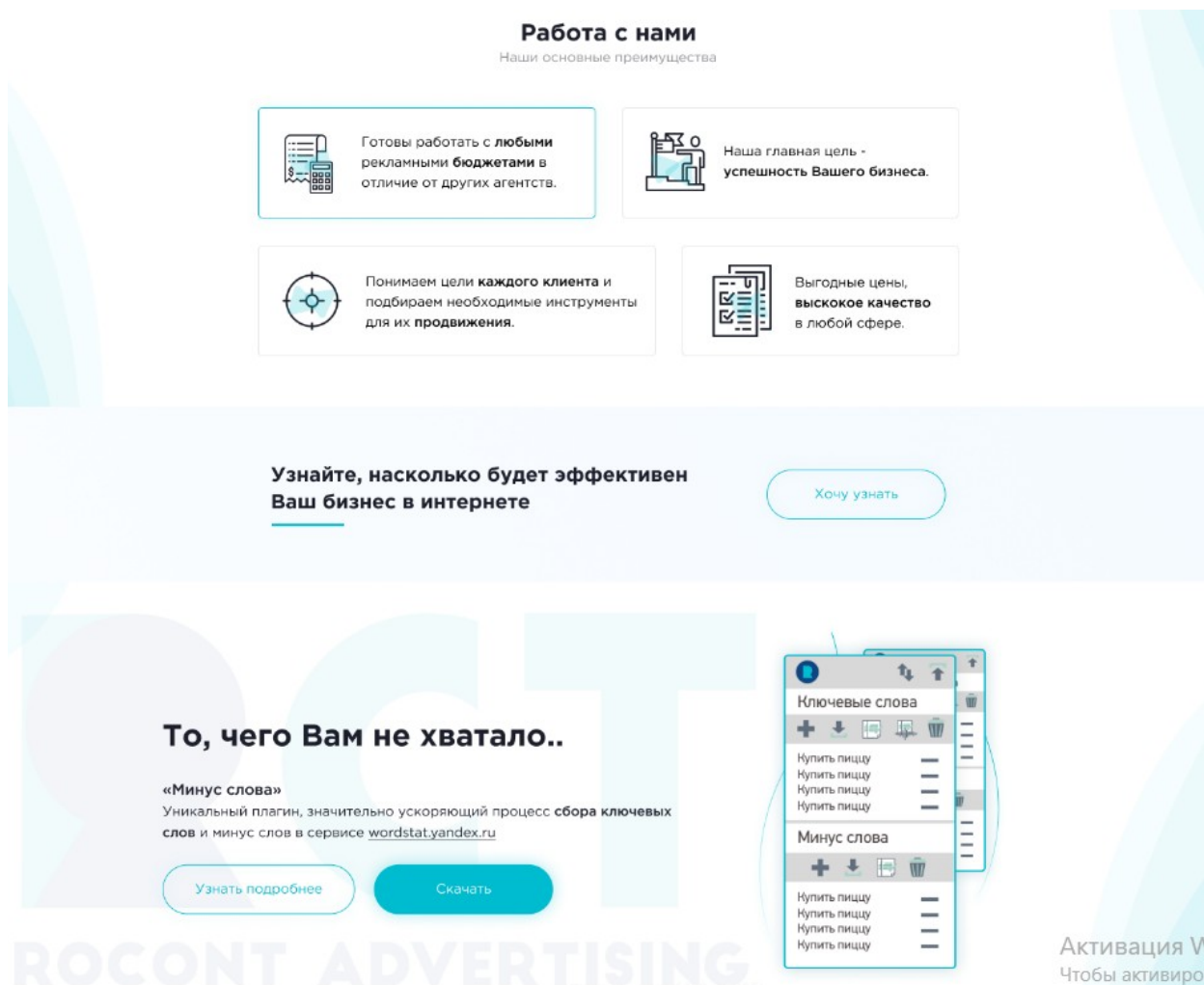


Рисунок 11.2 – Веб-дизайн главная страница, нижняя часть

Приступаем к веб-дизайну раздела возможности. Второй экран веб-дизайна приведен на рисунке 12.



Рисунок 12 – Веб-дизайн раздела возможности

Приступаем к веб-дизайну раздела портфолио. Третий экран веб-дизайна приведен на рисунке 13.

НАШЕ ПОРТФОЛИО

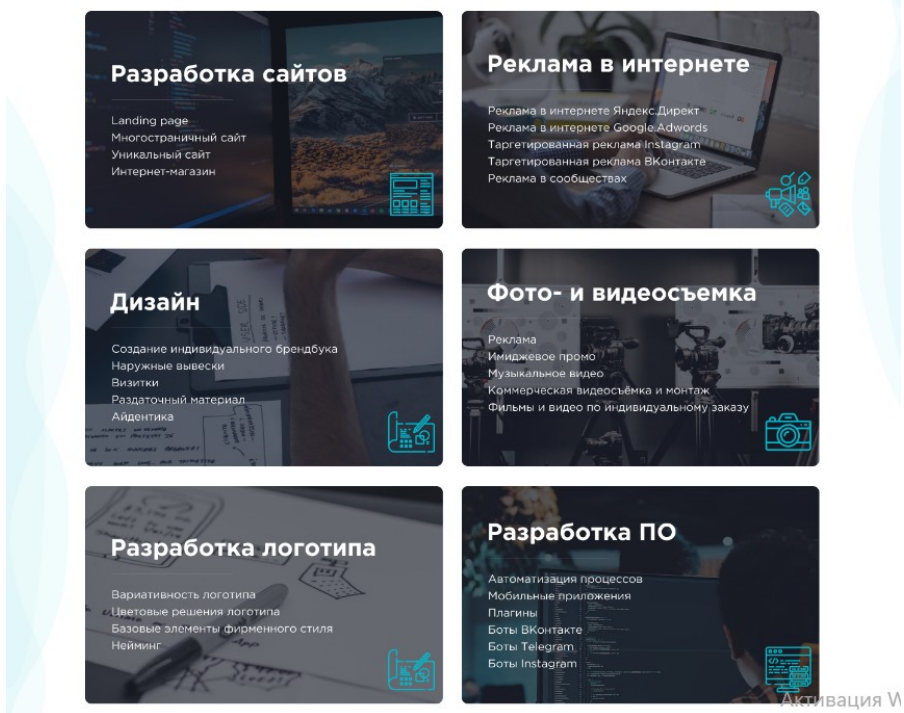


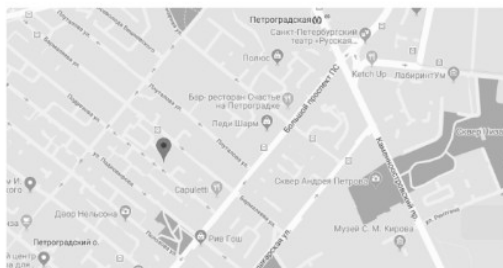
Рисунок 13 – Веб-дизайн раздела портфолио

Приступаем к веб-дизайну раздела контакты. Четвертый экран веб-дизайна приведен на рисунке 14.

КОНТАКТЫ

Адрес офиса в СПб:

Санкт-Петербург, БЦ Кантемировский



Информационная линия:

Городской номер: +7 (812) 629-6097

Мобильный номер: +7 (911) 211-78-10

info@rocont.ru



Обратная связь

Ответим на любой вопрос

Ваше Имя:

Иван

Ваш Email:

ivan@yandex.ru

Ваш Телефон:

79223114181

Ваш вопрос:

Напишите ваш вопрос

Отправить

Рисунок 14 – Веб-дизайн раздела контакты

2.3 Верстка веб-сайта

Перед непосредственным созданием веб-сайта необходимо подготовить папки в которых будут содержаться необходимые документы такие как шрифты изображения таблицы стилей библиотеки и скрипты. Также должна содержаться страничка index.html которая будет являться главной страницей веб-сайта. Структура папок веб-сайта приведена на рисунке 15.



Рисунок 15 – Структура папок веб-сайта

Для начала мы устанавливаем в файле index.html всю необходимую служебную информацию подключаем стили указываем титульную надпись веб-страницы также указываем путь до каскадной таблицы стилей. Все это делается в теге <head>. Служебная информация веб-страницы приведена на рисунке 16.

```
<head>
  <style type="text/css">...</style>
  <meta charset="UTF-8">
  <meta name="viewport" content="width=device-width, initial-scale=1">
  <meta name="keywords" content="рекламное агентство полного цикла, продвижение бизнеса онлайн, реклама в интернете цена">
  <meta name="description" content="Rocont Advertising - рекламное агентство полного цикла, которое занимается онлайн и оффлайн продвижением. Наша специализация - комплексное продвижение в интернете, брендрование и оффлайн реклама.">
  <title>Рекламное агентство | Санкт-Петербург | Rocont</title>
  <link rel="stylesheet" href="https://use.fontawesome.com/releases/v5.8.2/css/all.css" integrity="sha384-oS3vJWv+0UjzBFQzYUhtDYw+Pj2yciDJxpsK10YPAYqT085Qq/1cq5FLXAZQ7Ay" crossorigin="anonymous">
  <link rel="stylesheet" href="/assets/css/fonts.css" type="text/css">
  <link rel="stylesheet" href="/assets/css/style.css" type="text/css">
  <link rel="stylesheet" href="/assets/css/jquery.fancybox.min.css">
  <script async src="https://connect.facebook.net/en_US/fbevents.js"></script>
  <script async src="https://mc.yandex.ru/metrika/tag.js"></script>
  <script type="text/javascript">...</script>
  <noscript>...</noscript>
  <script type="text/javascript">...</script>
  <script type="text/javascript" async src="https://vk.com/js/api/openapi.js?160"></script>
  <noscript>...</noscript>
  <script>...</script>
  <noscript>...</noscript>
  <style type="text/css">...</style>
</head>
```

Рисунок 16 – Служебная информация веб-страницы

Также необходимо подключить шрифты, которые будут использоваться при верстке, которые будут использоваться по умолчанию. Подключение шрифтов приведено на рисунке 17.

```
1 ▼ @font-face {
2   font-family: 'Gotham Pro Regular';
3   src: url('../fonts/GothamPro.eot');
4   src: url('../fonts/GothamPro.eot?#iefix') format('embedded-opentype'),
5        url('../fonts/GothamPro.woff') format('woff'),
6        url('../fonts/GothamPro.ttf') format('truetype');
7   font-weight: normal;
8   font-style: normal;
9 }
10
11 ▼ @font-face {
12   font-family: 'Gotham Pro Medium';
13   src: url('../fonts/GothamPro-Medium.eot');
14   src: url('../fonts/GothamPro-Medium.eot?#iefix') format('embedded-opentype'),
15        url('../fonts/GothamPro-Medium.woff') format('woff'),
16        url('../fonts/GothamPro-Medium.ttf') format('truetype');
17   font-weight: normal;
18   font-style: normal;
19 }
20
21 ▼ @font-face {
22   font-family: 'Gotham Pro Bold';
23   src: url('../fonts/GothamPro-Bold.eot');
24   src: url('../fonts/GothamPro-Bold.eot?#iefix') format('embedded-opentype'),
25        url('../fonts/GothamPro-Bold.woff') format('woff'),
26        url('../fonts/GothamPro-Bold.ttf') format('truetype');
27   font-weight: normal;
28   font-style: normal;
29 }
30
31 ▼ @font-face {
32   font-family: 'Gotham Pro Black';
33   src: url('../fonts/GothamPro-Black.eot');
34   src: url('../fonts/GothamPro-Black.eot?#iefix') format('embedded-opentype'),
35        url('../fonts/GothamPro-Black.woff') format('woff'),
36        url('../fonts/GothamPro-Black.ttf') format('truetype');
37   font-weight: normal;
38   font-style: normal;
39 }
40
```

Рисунок 17 – Подключение шрифтов

Далее ориентируясь на макет начинаем создавать блоки с нужным содержанием внутри тега `<body>` и в файле `style.css` описывать свойства объектов. Полный текст HTML кода содержится в приложении А.

Все отдельные элементы веб-сайта удобно заключать в теги `<div>` и ``.

Тег `<div>` и тег `` не имеют свойств по умолчанию для внешнего отображения они представляют собой так называемые контейнеры поэтому к ним можно применять всевозможные CSS-стили чтобы элементы внутри этих тегов приняли желаемый вид.

В настоящее время тег `<div>` используется для группирования блочных элементов. Тег также позволяет объединять любой набор таких элементов как заголовок несколько абзацев список в единственном блоке который потом можно позиционировать на веб-страницы создавая сложную схему разметки. Тег `<div>` следует применять только в том случае если для выделения области не подходит другой элемент страницы. Тег `` применяется для выделения внутренних (inline) элементов таких как

отдельные слова и фразы, находящихся в пределах абзаца текста или заголовка.

Верстка происходит сверху вниз поэтому, следуя макету сперва создаем логотип, который также будет является ссылкой на главную страницу веб-сайта. Полный текст html-кода содержится в приложении А.

На веб-сайте имеется возможность клиенту оставить заявку на определенную услугу. Пример формы заказа товара приведен на рисунке 18.

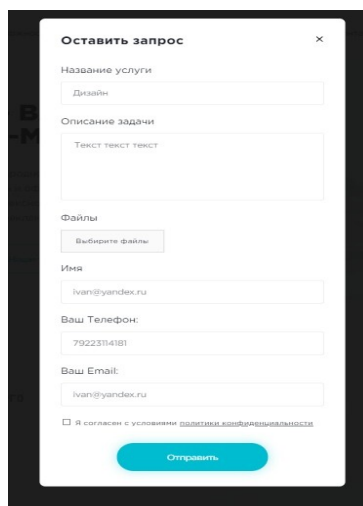


Рисунок 18 – Форма заказа услуги

При заполнении формы на электронную почту придет письмо с введенными данными, и сотрудники офиса могут связаться с клиентом.

Также на веб-сайте имеется возможность связаться с клиентом и обсудить детали, через форму обратной связи. Пример формы заказа товара приведен на рисунке 19.

Рисунок 19 – Форма обратной связи

При заполнении формы на электронную почту придет письмо с ответом на интересующий вопрос клиента по той или иной услуге.

Подвал веб-сайта содержит контактную информацию, а также в нем содержатся основные пункты меню и контактная информация. Также здесь располагается встроенная карта, на которой подробно можно посмотреть где находится офис компании. Пример подвала веб-сайта приведен на рисунке 20.

Карта сайта		Контакты	Адрес
О нас	3D визуализация	+7 (812) 629-6097	Санкт-Петербург, Подрезова ул. д5
Возможности	Фото и видеосъемка	info@rocont.ru	
Реклама в интернете	Портфолио		
Разработка сайтов	Контакты		
Дизайн			

Активация Windows
Чтобы активировать Windows, перейдите в раздел "Параметры".
© 2019 Rocont. Все права защищены.

Рисунок 20 – Подвал веб-сайта

Заключение

В настоящее время веб-сайт — это необходимая площадка каждой современной организации. Веб-сайт может считаться лицом компании в интернете. Ведь благодаря веб-сайту компании могут понятно доводить

информацию до партнеров и клиентов, структурированно излагать свою политику работы, а от этого зависит успех любого бизнеса. В рамках данной работы мы рассмотрели основные теоретические вопросы создания веб-сайтов, языковые и программные средства, используемые при разработке веб-сайтов. Нами были изучены современные методы создания веб-сайтов, от банальной ручной верстки сайтов и с использованием систем управления содержимым (CMS). В итоге мы разработали логическую структуру и веб-дизайн для рекламной компании «Roscont». Используя язык текстовой разметки HTML и каскадных таблиц стилей CSS. Впоследствии мы оптимизировали и минимизировали верстку для быстрого отображения в браузерах. Были прописаны определенные сео-теги, с помощью которых веб-сайт занимает хорошие позиции в поиске. Результатом проделанной работы является полноценный рабочий веб-сайт компании «Roscont» наполненный информационным контентом.

Список используемых источников

1. Гончаров А.Н. Самоучитель HTML А.Н. Гончаров.- СПб.: Питер, 2012.
2. Горнаков С.Г. Осваиваем популярные системы управления сайтом С.Г. Горнаков.- М.: Наука, 2015.
3. Ежевский Д.О. О критериях создания электронных учебных пособий Д.О. Ежевский.- М.: Эксмо, 2014.
4. Колисниченко Д.Н. Движок для вашего сайта. CMS Joomla!, Slaed, PHP-Nuke Д.Н. Колесниченко.- М.: ИТ Пресс, 2016.
5. Колисниченко Д.Н. Joomla!: Руководство пользователя / Д.Н. Колисниченко.- М.: Диалектика, 2017.
6. Норт Б. Joomla! : Практическое руководство / Б. Норт.- М.: Символ-плюс, 2015.
7. Рамел Д. Самоучитель Joomla! / Д. Рамел.- СПб.: Питер, 2015.
8. Хаген Г. Создание веб-сайтов с помощью Joomla! 1.5 / Г. Хаген.- М.: Вильямс, 2015.
9. Чиртик А.В. Популярный самоучитель HTML / А.В. Чиртик.- СПб.: Питер, 2016
10. Шишкин В.В. Многоуровневая концепция создания электронных учебных пособий / В.В. Шишкин. - М.: Махаон, 2014.

