

**Министерство образования Московской области**

**Государственное бюджетное профессиональное образовательное учреждение Московской области**

**«Щелковский колледж»**

**(ГБПОУ МО «Щелковский колледж»)**

ДИПЛОМНЫЙ ПРОЕКТ

по специальности

**09.02.07. «Информационные системы и программирование»**

*(код, наименование специальности)*

Тема: «Разработка сайта-визитки для фотостудии «Фото+» г. Королев ».

|  |  |
| --- | --- |
|  | Студент/ка  Агаджанян Норик Оганнесович  Группа 919  Руководитель ВКР  Морозова Алевтина Викторовна  Консультанты\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_  \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ |
|  | Работа защищена «\_\_\_\_» \_\_\_\_\_\_\_\_\_\_\_\_2022 г.  с оценкой «\_\_\_\_\_\_\_\_\_\_\_\_\_»  Секретарь ГЭК: \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_/ /  подпись ФИО |

# СОДЕРЖАНИЕ

[ВВЕДЕНИЕ 3](#_Toc99736609)

[1. **Глава 1. Зачем нужен сайт визитка и кому он подходит** 5](#_Toc99736610)

[1.1. История возникновения и создание визитных карточек 5](#_Toc99736611)

[1.2. Что такое сайт визитка 7](#_Toc99736612)

[1.3. Особенности сайта визитки 9](#_Toc99736613)

[1.4. Кому и ачем нужен сайт-визитка.Его преимущества и недостатки 10](#_Toc99736615)

[1.5. Структура типового сайта-визитки 13](#_Toc99736616)

[2. **Глава 2. Структура и принципы разработки сайта** 19](#_Toc99736617)

[2.1. Языки, используемые в процессе разработки сайта-визитки 19](#_Toc99736618)

[2.2. Основные принципы разработки сайта 34](#_Toc99736619)

[3. **Глава 3. Разработка сайта-визитки для фотостудии «Фото+» г.Королев** 39](#_Toc99736621)

[3.1. Правила создания сайта 39](#_Toc99736622)

[3.2. Стили 42](#_Toc99736623)

[3.3. Разработка сайта 46](#_Toc99736624)

[3.4. Основы техники безопасности сайта 53](#_Toc99736626)

[ЗАКЛЮЧЕНИЕ 59](#_Toc99736628)

[Список использованных источников 60](#_Toc99736629)

Введение

На сегодняшний день, трудно представить жизнь без доступа к сети интернет. Его развитие было столь стремительно, что уже спустя почти пятьдесят лет каждый желающий смог создавать web-сайты, используя их со своей целью: продвигать их на мировом рынке «спроса и предложения», сопровождать клиентов из разных городов и стран, создавать познавательные или развлекательные страницы.

На данный момент большинство современных организации, предоставляемых какие-либо услуги, уже невозможно представить без применения информационных технологий. Иметь собственный ресурс в интернете – это хорошая возможность кратко и ёмко заявить о себе на просторах всемирной паутины и привлечь клиентов. Сегодня для успешной рекламной компании важно иметь свою страницу в сети интернет, с описанием предоставляемых услуг. Владельцы малого бизнеса или стартаперы создают (заказывают) небольшой сайт, где размещены: адрес, контакты, фотогалерея, небольшое описание данного вида услуг, новости, более крупные компании и холдинги имеют полноценный сайт с вариантами выбора или подбора информации, полное описание предлагаемой продукции/услуг, информации об истории и/или составе компании и пр.

Создание web-страницы ориентировано, прежде всего, на повышение эффективности работы организации. В нашей стране количество фотографов и фотостудий давно превышает спрос на фотоуслуги. Дорогостоящая профессиональная фототехника сейчас является доступной большому количеству людей, а онлайн-курсы помогут новичкам перенять опыт профессиональных фотографов, получить нужные знания и профессиональные навыки. Именно это и приводит к тому, что рынок фото услуг перенасыщен. Нестабильный спрос на услуги и высокая конкуренция побуждают фотографов искать разные способы привлечения потока клиентов, а наиболее распространенным является проведение тематических фотосъемок, на которых фотограф находит определенное количество желающих людей и делает снимки в определенных интерьерах, с набором костюмов, аксессуаров, декораций. Далеко не каждый фотограф может позволить себе свою собственную фотостудию, в связи с этим большинство фотографов пользуются услугами аренды фотостудии, что является прибыльной отраслью фото индустрии.

Актуальность web-сайта обусловлена ее оперативностью, масштабностью, подачи информации для широкого круга пользователей, возможностью оставлять отзывы, привлечением клиентов, осуществлением связи с экспертами из всевозможных уголков мира, возможностью опросов и т.д.

Предмет разработки: Сайт-визитка для фотостудии «Фото+ г. Королев».

Цель: Разработка web-сайта для фотостудии «Фото+ г. Королев».

Для достижения поставленной цели необходимо решить следующие задачи:

1. Произвести анализ средств разработки web-сайтов;
2. Определить основные элементы сайта;
3. Выбрать средства реализации для разработки сайта фотостудии;
4. Разработать сайта-визитки.

Данная работа состоит из введения, трех глав, заключения, списка литературы (списка использованных источников) и приложения.

# Глава 1. Зачем нужен сайт-визитка и кому он подходит

## 1.1. История возникновения и создания визитных карточек.

Визитки – обязательные аксессуары современного делового человека, разделяемые на персональные, корпоративные и деловые. В зависимости от типа, визитные карточки содержат краткую информацию о персоне, компании, товарах и услугах. Их используют для налаживания кратковременных и длительных бизнес-связей, самопрезентации, в качестве рекламного инструмента для продвижения товаров и услуг. Но возникли визитки не сегодня и даже не вчера. Их история насчитывает более 2500 лет.

Прототип визитной карточки впервые появился в средневековом Китае. Она представляла собой бамбуковую полоску или кусочек дерева, декорированный шелком. Позднее визитки распространились в Европе, откуда это изделие пришло в Россию. Визитные карточки, расписанные вручную художником, отражали статус владельца, коим были знаменитости, государственные люди и особы, имеющие дворянское происхождение.

Многовековая история визитной карточки очень интересна. Самые первые изделия появились более 25 веков назад в провинциях Китая. Предназначение у такой визитки было несколько иное: на ней писали иероглифы, повествующие о цели визита и прошение от имени владельца визитки. Изготавливали визитные карточки каллиграфы – специально обученные люди. Поскольку карточки были индивидуальные, универсального характера они не носили.

Чуть позднее при изготовлении визиток стали использовать шелк. Материал этот стоил больших денег, поэтому визитками пользовались преимущественно зажиточные люди.

Со временем вид карточек менялся, но предназначение их оставалось прежним. И только в 17 веке на территории Франции стали использовать визитки, схожие по исполнению с современными. Правила этики визитов при дворе Людовика предписывали порядок их применения. Однако высокая цена ограничивала распространения визитных карточек. Теперь это не были деревянные таблички, а над созданием визитки трудились художники, превращающие изделие в шедевр искусства.

Активное распространение визитные карточки получили в 19 веке. Взять, к примеру, Москву. Визитные карточки простых людей и вельмож существенно отличались. Люди низших сословий пользовались общими листами для визитов, а для чиновников предназначались листы, оставленные в фойе. Любой желающий мог прийти и поздравить своего начальника, оставив запись в подобном листе.

Несмотря на столь длинный и разнообразный путь развития, в разных словарях трактовка слова «визитка» одинаковая. Это карточка, содержащая информацию об ее владельце. Существуют уточнения по поводу размеров карты и материала, из которого ее изготавливают. Постепенно визитная карта вошла в правила этики в разных странах.

В 20 веке визитки стали активно использоваться в деловых кругах. В это время при оформлении визитки большое внимание уделялось ее внешнему виду, чем содержанию. В моде были красивые тесненные узоры, декор резьбой.

Напечатанный вариант визитки был обнаружен в Германии и датировался он 1786 годом. В современном исполнении визитки содержится реклама, а также контактная информация для связи. С давних времен сохранилась тенденция, когда визитка человека с высшим статусом отличается своим представительным оформлением.

Даже теперь, в век современных технологий, визитка продолжает выполнять основную свою функцию – сообщает информацию о своем владельце. Естественно, ручной труд в изготовлении визиток уже практически не применяется.

Активные, деловые и успешные люди непременно используют стильные визитки. Это наиболее практичный и удобный способ общения, который содержит в себе необходимую информацию о владельце такой карточки и его бизнесе.

## 1.2. Что такое сайт-визитка

Сайт-визитка — небольшой [сайт](https://ru.wikipedia.org/wiki/%D0%A1%D0%B0%D0%B9%D1%82), как правило, состоящий из одной (или нескольких) [веб-страниц](https://ru.wikipedia.org/wiki/%D0%92%D0%B5%D0%B1-%D1%81%D1%82%D1%80%D0%B0%D0%BD%D0%B8%D1%86%D0%B0) и содержащий основную информацию об организации, частном лице, компании, товарах или услугах, [прейскуранты](https://ru.wikipedia.org/wiki/%D0%9F%D1%80%D0%B5%D0%B9%D1%81%D0%BA%D1%83%D1%80%D0%B0%D0%BD%D1%82), контактные данные.

Одностраничный сайт с основной информацией о компании — это универсальное решение для бизнеса, нуждающегося в представлении в сети, но не желающего тратить много времени и ресурсов на создание полномасштабного корпоративного сайта.

Сайт-визитка предназначен для предоставления краткой информации о компании, ее услугах и товарах. Такой сайт позволяет не только обозначить свое присутствие в сети, но и получать обратную связь, собирать данные потенциальных клиентов, привлекать трафик посредством контекстной рекламы и популяризации в социальных сетях.

Когда компания или частное лицо не нуждается в большом подробном сайте, или, когда бюджет или время для создания сайта ограничены, оптимальным выбором станет **сайт визитка**.

Сайт визитка – это полновесный веб-ресурс, на страницах которого размещена основная информация об организации, компании или частном лице. По аналогии с визитными карточками, сайт визитка содержит краткую, но при этом исчерпывающую информацию об услугах, товарах и контактных данных фирмы или частного лица. Главное преимущество такого сайта – невысокая стоимость и оперативность изготовления. При этом грамотно оформленный сайт визитка эффективно выполняет рекламные функции.

Существует два основных типа сайтов-визиток: персональный сайт визитка и деловой (Рисунок 1).

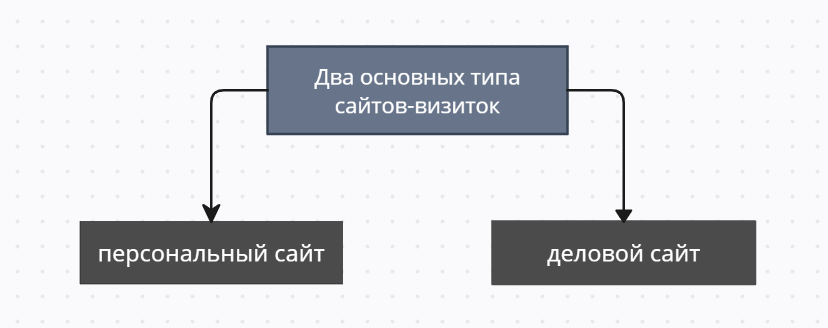


Рисунок 1. Два основных типа сайтов-визиток

Персональный сайт визитка преимущественно создается для определенного круга лиц – родственников, друзей, знакомых. Он может содержать личную информацию о владельце сайта, различные записи (например, стихотворения собственного сочинения), фотографии (семейные, детские) и т.д.

Деловой сайт визитка полезен не только для представления компании или фирмы в Сети, но и для частных предпринимателей или «вольных художников» (дизайнеров, фотографов, копирайтеров). Он содержит всю основную информацию о деятельности фирмы или частного лица и позволяет посетителям ознакомиться с перечнем товаров и услуг или с портфолио владельца сайта.

В зависимости от разновидности, ресурсы отличаются количеством страниц и объемом информации. Все разновидности можно использовать для продвижения товаров или услуг в интернете.

Также сайты визитки могут быть статическими и динамическими (Рисунок 2).



Рисунок 2. Виды сайтов-визиток

Статический сайт визитка – это простейший сайт, который несет стабильную информацию о фирме (контактные данные или юридические реквизиты) и не требует регулярных изменений контента. Так как информация постоянна, такой вид сайта не нуждается в поддержке и сопровождении.

Динамический сайт – это сайт, содержащий такие функции, как обратная связь, модуль новостей и т.д. Контент такого сайта периодически обновляется, что позволяет регулировать актуальность размещенной информации.

CMS – это система управления контентом сайта, или, как ее еще называют, «движок сайта». Эта специальная программа устанавливается на хостинг-площадке (платной или бесплатной) и выполняет две основные функции:

1. Отображение страниц сайта, которое генерируется непосредственно в момент запроса пользователя. При этом пользователю может быть показана уникальная информация, предназначенная лично для него (например, содержимое его персональной корзины в Интернет-магазине).
2. Управление сайтом (добавление новых страниц, публикация новых видео, фото и т.д.). Для редактирования страницы сайта владельцу или администратору сайта доступен специальный визуальный редактор, при помощи которого можно форматировать текст, добавлять ссылки или картинки и при этом сразу видеть, как новая информация будет размещена на сайте.

## 1.3. Особенности сайта-визитки

Цель визитки, являющейся электронным аналогом визитки картонной, краткая самопрезентация с предоставлением контактных данных, направленная на наращивание положительной репутации.

Размеры сайта-визитки определяются объемом информационных сведений. Это может быть как одна страница на два экрана, так и небольшой ресурс на 10–15 страниц. В последнем случае сайт также содержит статичную информацию, не требующую постоянной актуализации (контактные данные, описание товаров/работ/услуг и т.п.). В дальнейшем возможно использование визитки для создания полноценного корпоративного ресурса.

Электронная визитка имеет следующие особенности:

* ненавязчивый скромный дизайн, не отталкивающий потенциальных клиентов;
* небольшой объем (до 20 страниц);
* контент, не требующий постоянного обновления:
  + краткое описание товаров/работ/услуг, дающее представление о характере деятельности владельца визитки;
  + контактные данные (телефоны, адреса, схема проезда, ссылки на соцсети и проч.) для связи. Владелец компании, уже имеющей историю, может ее также опубликовать;
* минимум или отсутствие возможностей (интерактивные виджеты, личный кабинет, чат-бот и т.п.) для действий пользователя. Возможно размещение формы для заказа телефонного звонка.

## 1.4. Кому и зачем нужен сайт-визитка. Его преимущества и недостатки

Электронная визитка, не подразумевающая взаимодействие с пользователем онлайн, является оптимальной формой презентации для самозанятых специалистов, стартапа и небольших компаний. С ее помощью осуществляют бизнес-продвижение:

* малоизвестные компании на начальных стадиях предпринимательства и раскрутки бренда. Со временем, когда молодой бизнес сможет позволить себе увеличения бюджета на рекламу, он преобразует сайт-визитку в полноценный ресурс;
* небольшие и средние компании, частные самозанятые лица, указывающие геопривязку и оказывающие услуги оффлайн (торговля, строительство, ремонт, клининг, юриспруденция, бухгалтерия, педагогика, бьюти-сфера и проч.);
* компании и частные лица, оказывающие услуги онлайн (программирование, дизайн, перевод, копирайт и т.п.), визитки которых могут содержать ссылки на выполненные работы (портфолио).

Электронные визитки выигрывают у своих картонных предшественников, рассказывая более полно о деятельности их владельцев и продукте, производимом ими.

На интернет-странице вы разместите намного больше информации, чем на обычном кусочке картона размером 90х50 мм. При этом, информация будет все еще лаконичной, но лучше запомнится вашему потенциальному покупателю или партнеру.

Сайт можно легко найти в интернете, таким образом, он выступает дополнительной рекламой. А визитку можно получить только оффлайн от владельца компании или сотрудников, что существенно сокращает потенциальную целевую аудиторию.

Это универсальный веб-ресурс, который, на самом деле, подходит практически всем:

* Малому бизнесу
* Частным лицам (специалистам в своем деле)
* Предприятиям, которые оказывают услуги
* Бизнесам, которым подходит корпоративный сайт, но нет возможности его создать

Сайты-визитки универсальны своим содержанием и формой. Ведь каждый предприниматель рассказывает о своей компании, услугах и оставляет свои контактные данные. Поэтому подобные онлайн-визитка точно беспроигрышный вариант для вас, если нет потребности в интернет-магазине или полноценном корпоративном сайте.

И все же, создание веб-ресурса требует времени, терпения и финансовых вложений. Предлагаем рассмотреть все преимущества и недостатки сайта-визитки, чтобы понять есть ли в нем необходимость.

Преимущества:

* Лаконичная, но емкая презентация компании, личности или продукта
* В случае необходимости можно масштабировать до полного корпоративного сайта
* Привлечение клиентов в интернете, а не только оффлайн
* Оптимальная стоимость создания
* Быстрые сроки разработки
* Простота в обслуживании
* Можно использовать как портфолио для фрилансеров и специалистов в любой сфере

Недостатки:

* Упрощенный функционал сравнительно с многостраничными сайтами
* Подходит не всем компаниям по корпоративному статусу
* Не все понимают, что такое сайт визитка, и могут воспринять как просто малоинформативный сайт

Начнем с того, что в наше время сайт необходим каждой компании. Он отлично привлекает клиентов через интернет и помогает расти вашему бизнесу. Но надо ли вам именно сайт- визитка?

Так как главная цель такого сайта лишь представить компанию или личность в интернете, то все же преимущества выше весомые. От такого сайта лучше не ждать бешеного роста продаж, ведь он только информирует заинтересованных о компании, а не стимулирует желание купить офферами и сумасшедшими скидками.

Нужен ли вам сайт-визитка? Однозначно да, вы хотите лаконично заявить о себе вашей целевой аудитории в интернете. Но если вы ждете продажи с сайта или хотите большой многостраничный сайт – это точно не ваш вариант.

### 1.5. Структура типового сайта-визитки

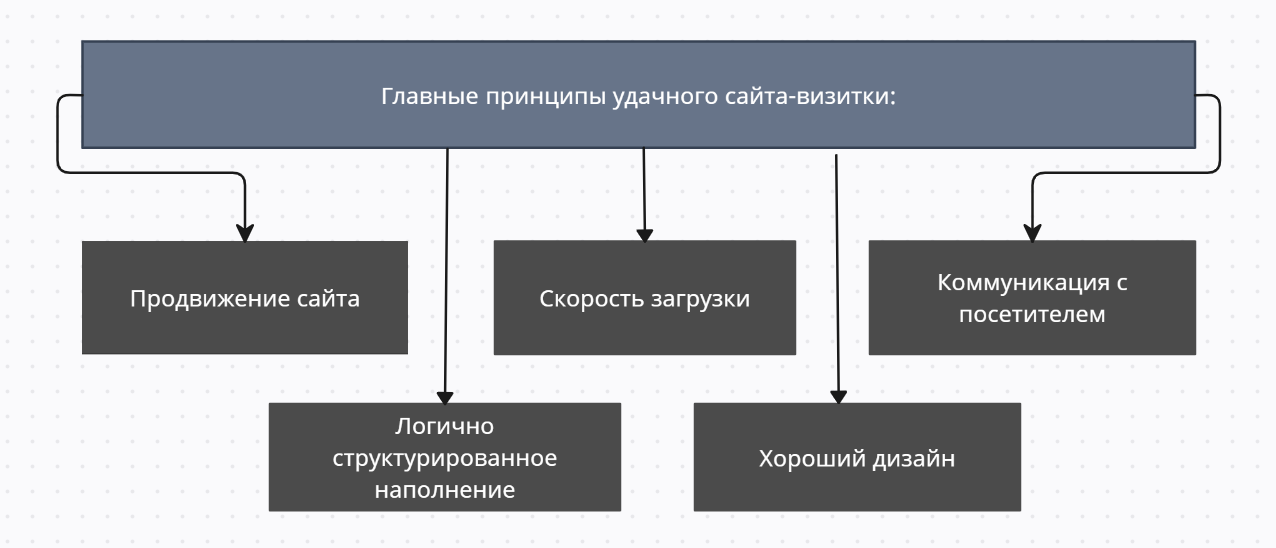
При создании сайта-визитки обязательно помните про принцип “чем проще – тем лучше”. Не всегда оригинальный дизайн сыграет на руку (Рисунок 3).

Рисунок 3. На что обратить внимание во время создания сайта-визитки

Главные принципы удачного сайта-визитки:

* Хороший дизайн.

Текст должен быть читабельным, а глаза посетителя не должны уставать от ярких цветов и картинок. Лучше всего создать сайт в фирменном стиле компании, избегая сочетания более, чем 3 цветов и множества шрифтов.

* Логично структурированное наполнение.

Посетитель должен интуитивно понимать и быстро находить нужную информацию. Структурируйте тексты, выделяйте заголовки, а контактную информацию располагайте в порядке приоритетности (если адрес номера телефона, то расположите его на первом месте).

* Коммуникация с посетителем.

К примеру, если человек оставил данные в контактной форме, то оставьте для него сообщение “мы скоро свяжемся с вами”. А если он попал на страницу 404, то оставьте гиперссылку на главную страницу сайта.

* Скорость загрузки.

Вес и количество изображений, анимированный дизайн, вес сайта и хостинг – это все влияет на скорость. Скорость должна быть около 2 секунд, а проверить ее можно в сервисе Google PageSpeed Insight.

* Продвижение сайта.

Так как на сайте-визитки мало текстов, SEO продвижение слегка затрудняется, но возможно, как минимум по брендовым запросам компании. Уделите особое внимание составлению семантического ядра и перелинковке на сайте.

Также, отличный результат приносит контекстная реклама. Настройте ее с привлекательным рекламным обращением на целевую аудиторию. Так ваш сайт будут посещать заинтересованные люди.

Структура любого сайта-визитки напрямую связана с его функцией. На нем должна содержаться вся информация о товарах и услугах компании, контакты. Этот сайт является аналогом обычной картонной визитки, на которой указаны название компании, ФИО директора, телефоны. Как и картонная визитка, сайт имеет небольшой размер. Но, несмотря на это, на нем должна быть выложена вся необходимая информация. Если сайт будет качественным, то он сможет привлечь новых клиентов, соответственно, компания получит прибыль. Это важная своеобразная реклама.

Для создания сайта-визитки существует определенный шаблон, который включает в себя несколько пунктов. Это создание самой модели, цветовое оформление, наполнение его всей необходимой информацией, публикация данного ресурса в Интернете.

На первом этапе создается модель сайта-визитки. Любая недорогая веб студия в Москве предложит десятки вариантов визитки. Это обычный сайт со страницами и переходами, только без информации, текста и картинок. Для того чтобы создать модель, нужно четко представлять, из скольких страниц будет состоять сайт.

На главной странице должно быть размещено приветствие и краткая информация о предприятии, на следующей – вся информация о товаре с фотографиями и прайс-листом.

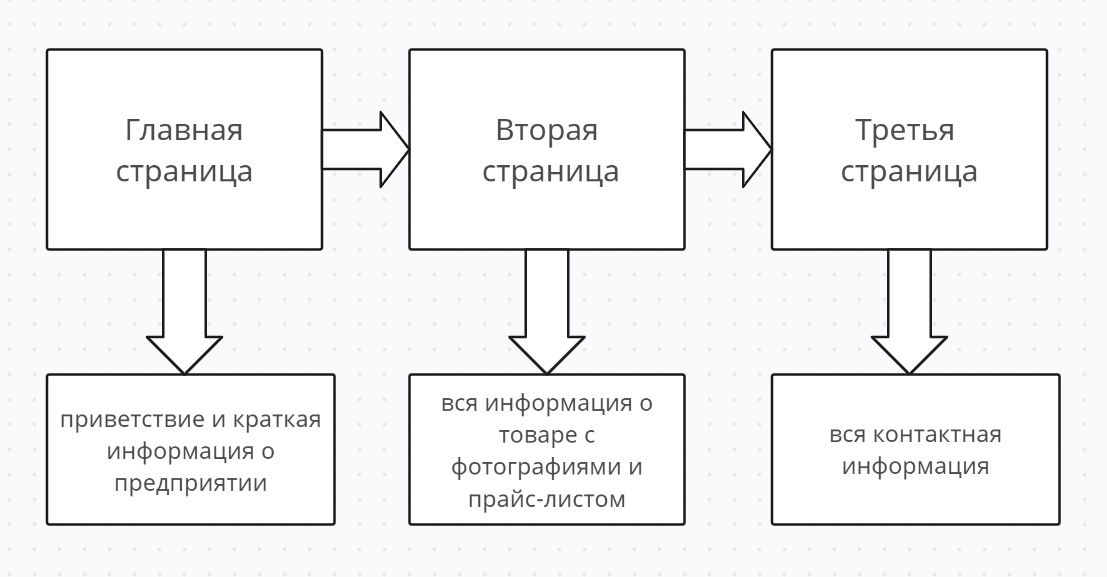
На третьей странице нужно поместить всю контактную информацию (Рисунок 4).

Рисунок 4. Модель сайта визитки

На этапе цветового оформления нужно создать фон, картинку для обоев. Можно задать так же фоновую музыку, которая будет звучать при открытии страницы. Все картинки можно брать из галереи, можно загружать и качественные фотографии.

Дизайн сайта – визитки должен быть уникальным. Тогда он сможет привлечь к себе больше внимание. В этом вопросе нет никаких ограничений. Можно сотворить все что угодно. Главное, чтобы фантазия работала в нужном направлении.

Публикация сайта - не такая уж и страшная процедура, как кажется на первый взгляд. Нужно правильно подобрать хостинг. Он может быть как платным, так и бесплатным. Но сайт-визитка не требует особой раскрутки и продвижения по поисковым системам. Поэтому вполне достаточно будет и бесплатного хостинга.

Чтобы как-то продвинуть сайт-визитку, можно его адрес указать на обычных картонных визитках, разослать коллегам, партнерам по электронной почте. Интернет является неоценимым ресурсом. Если ваша компания или предприятие заявит о себе именно на страницах всемирной Сети, то это будет огромным плюсом. Ведь информацию могут прочитать многие люди, и не исключено, что именно благодаря сайту-визитке они станут вашими клиентами.

### Шапка + меню

Это первое, на что обращает внимание посетитель сайта в поисках информации. Обязательного стандарта в оформлении нет, мы размещаем на наших сайтах-визитках такие элементы: логотип, название компании, дескриптор, адрес, контактные данные (номера телефонов, кнопки мессенджеров или социальных сетей, адрес электронной почты, кнопки обратной связи).

Если у вас нет логотипа, и вы задумываетесь о его разработке, советуем сначала прочитать нашу статью "Когда приступать к разработке логотипа и почему это может подождать?"

Если логотип есть, но не отображает, чем занимается ваша компания, на помощь придет дескриптор. Дескриптор - это краткое описание деятельности компании. Он может быть под или рядом с логотипом или названием.

Под шапкой располагается меню по разделам сайта. Оно должно быть простым, логичным и понятным.

### Первый экран

Первый экран - то, что находится под меню. На наших сайтах-визитках здесь размещают Уникальное Торговое Предложение (УТП) или информацию об акциях, конкурсах и розыгрышах призов.

Уникальное торговое предложение сообщает об особых свойствах, выгодах товара или услуги для целевой аудитории. Должно отвечать трем простым требованиям:

* быть понятным (говорите на одном языке с вашей аудиторией),
* быть конкретным (детализируйте, преподносите ясные выгоды, но не перегружайте информацией),
* быть уникальным (покажите выгодные отличия от конкурентов, ценность вашего продукта для клиентов).
* Остальные блоки можно менять в зависимости от информации, которая у вас есть. Из них строится **второй экран -** самая большая область, где размещают текстовый, графический, видео и аудио контент. Одни и те же блоки можно использовать под разные задачи.
* Чаще всего под первым экраном размещают **преимущества** компании, товара или услуги. Для этого используют блок с иконками, блок с текстом, блок текст + изображение. По желанию клиента, блоки можно расположить в другом месте.

Под преимуществами располагают информацию о компании, её составе, руководстве. Делают это с помощью блоков:

* Заголовок + текст + изображение;
* Заголовок + текст;
* Текст + изображение;
* Видео.

Основой сайта визитки, конечно же, является презентация товаров или услуг компании. У нас есть несколько блоков, чтобы представить их красиво, полно и разнообразно:

* блок галерея;
* блок текст + изображение;
* блок с видео;
* блок с аккордеоном.

Также на сайте-визитке часто описывают **этапы работы** или этапы заказа. Для этого используют такие блоки: иконки, текст, текст + изображение.

Для удобства клиентов подрядчики размещают стоимость услуг или товаров, а также условия оплаты в целом. Мы предлагаем несколько вариантов того, как это можно сделать:

* с помощью текста;
* с помощью текста и изображения;
* с помощью таблиц;
* с помощью прикрепленных файлов. Такой вариант используется, если нужно прикрепить прайс целиком или наименований товаров слишком много.

Для размещения документов, содержащих техническую, юридическую, нормативную или иную документацию используют обычно два блока: либо галерею изображений, либо прикрепленные файлы.

## Подвал

Подвал - это самая нижняя часть страницы, куда выносится полезная, но не первостепенная информация. Здесь размещают: авторские права, формы захвата и формы связи, контактные данные (номера телефонов, электронную почту, кнопки мессенджеров, адреса офисов и их график работы, карту проезда).

Как видим, типовой сайт-визитка имеет простую и понятную структуру и собирается менеджером из разных блоков в зависимости от информации заказчика.

Сайт-визитка является отличным решением для многих бизнесов и частных лиц, которым необходимо разместить информацию о себе в интернете и привлечь новых клиентов.

Глава 2. Структура и принципы разработки сайта

2.1 Языки, используемые в процессе разработки сайта-визитки

Для разработки сайта, я использовал текстовый редактор Visual Studio Code, я а в качестве языков – Html, CSS, JavaScript, Php.

Для начала разберем, что такое язык программирования.

Язык программирования – это формальная знаковая система, на которой пишут компьютерные программы. Его можно представить как набор разнообразных правил и команд, на основе которых программист пишет код.

Язык программирования - средство общения между разработчиком и компьютером. Программист пишет код, который говорит устройству о том, что ему нужно сделать, какой процесс выполнить. Сама последовательность действий называется [алгоритмом](https://blog.skillfactory.ru/glossary/algoritm/), а язык программирования помогает ее записать.

Visual Studio Code (VS Code) — это редактор [кода](https://blog.skillfactory.ru/glossary/programmnyj-kod/) для разных [языков программирования](https://blog.skillfactory.ru/glossary/yazyk-programmirovaniya/). Он относительно немного весит, гибкий и удобный. В нем можно писать, форматировать и редактировать код на разных языках.

Visual Studio Code не стоит путать с Visual Studio — это [IDE](https://blog.skillfactory.ru/glossary/ide/), очень мощная и масштабная, но одновременно с этим тяжеловесная. Названия похожи, потому что у обоих средств разработки один и тот же создатель, но продукты разные по своей сути. Visual Studio Code меньше весит, проще в освоении и подходит в том числе для начинающих разработчиков.

VS Code позволяет легко писать, форматировать и редактировать код на разных языках. С его помощью можно быстро создать проект и структуру файлов в нем, он подсвечивает [синтаксис](https://blog.skillfactory.ru/glossary/sintaksis/) кода и помогает автоматически править ошибки. В нем есть возможности для [отладки](https://blog.skillfactory.ru/glossary/otladka-debugging/) и запуска кода на некоторых языках.

Редактор легко расширяется, поэтому к перечисленным функциям можно добавить новые - достаточно просто скачать нужное дополнение из официального каталога. Дополнения тоже распространяются бесплатно.

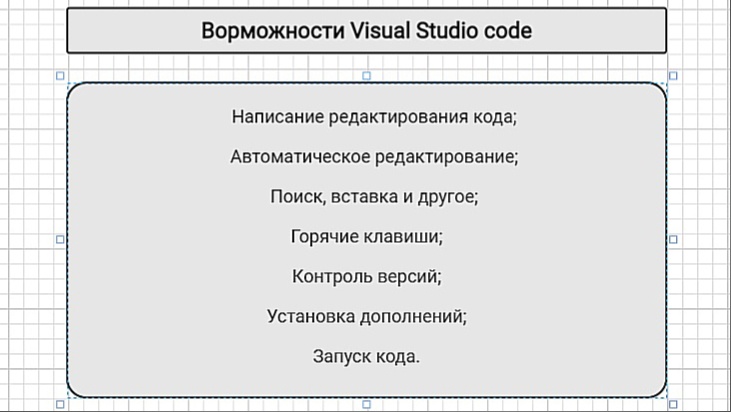


Рисунок 5. Возможности Visual Studio Code

Написание и редактирование кода. Первое и главное назначение редактора — писать исходный код на каком-то из языков программирования, редактировать и сохранять его. Так же как редакторы текста оснащены функциями для форматирования контента, редактор кода позволяет структурировать и форматировать программы. Например, он подсвечивает синтаксис, выделяет ошибки, автоматически расставляет отступы. В нем есть и автодополнение: если ввести первые несколько букв команды, он предложит варианты, как можно ее закончить. В VS Code есть целая система для помощи в редактировании кода, ее компоненты сгруппированы под общим названием IntelliSense.

Автоматическое редактирование. Редактирование и исправление кода в VS Code можно автоматизировать. Существуют специальные плагины и расширения для редактора, которые автоматически исправляют мелкие ошибки в программе: неправильные отступы, именование, несоответствие стилю кода и так далее. Кстати, стиль кода можно задать в настройках. Конечно, сложную ошибку такой модуль не исправит: он не поймет, как это можно сделать. Но для устранения мелких недочетов возможность удобная.

Поиск, вставка и другое. Среди возможностей Visual Studio Code — быстрая навигация по документу и его редактирование. Например, строка поиска поддерживает в том числе регулярные выражения — формулы для разных текстовых сочетаний. С редактированием тоже удобно: можно написать сокращенную формулу той или иной команды, и редактор достроит ее сам. А можно, например, выделить код и закомментировать его одним сочетанием клавиш.

Горячие клавиши. Горячие клавиши позволяют применять те или иные функции, не заходя в меню. Достаточно просто нажать нужное сочетание на клавиатуре. В VS Code таких клавиш много, и их можно перенастроить под свои вкусы — как вам удобнее.

Контроль версий. VS Code «из коробки» поддерживает систему контроля версий [Git](https://blog.skillfactory.ru/glossary/git/" \t "_blank). Прямо из редактора можно закоммитить изменения или откатить коммит, добавить к нему комментарий или отправить в удаленный [репозиторий](https://blog.skillfactory.ru/glossary/repozitorij/" \t "_blank). Для этого не нужно входить в консоль и набирать команды: все делается через [визуальный интерфейс](https://blog.skillfactory.ru/glossary/gui/). Сделать коммит можно в несколько кликов в специальной боковой панели.

Установка дополнений. Одна из особенностей VS Code — он очень легко расширяется. Прямо из редактора можно перейти в каталог расширений и дополнений, найти в поиске то, что вам нужно, и установить. Так подключаются поддержка других языков, плагины для автоматического исправления, конфигураторы и много чего еще. Благодаря легкой расширяемости VS Code можно полностью настроить под себя и свои нужды.

Запуск кода. Изначально редактор позволяет запускать код только для малого количества языков. Но к нему есть дополнения, которые открывают такую возможность и для других. Чтобы воспользоваться ей, нужно установить соответствующее расширение и перезагрузить VS Code. После этого в интерфейсе редактора должны появиться кнопки для сборки и запуска. Кстати, изначально в нем есть целая вкладка под названием «Запуск и отладка».

Преимущества Visual Studio Code

* Простота освоения. У редактора интуитивно понятный интерфейс, его легко установить и настроить. А после этого им сразу можно начинать пользоваться. Единственной проблемой поначалу могут стать горячие клавиши и дополнительные настройки — они необычно организованы. Но со временем легко разобраться, особенно если вначале держать под рукой памятку или пользоваться туториалами.
* Малый вес. По сравнению с IDE редактор быстро работает и мало весит, не так требователен к ресурсам компьютера. При этом он довольно функционален, а огромное количество расширений позволяет добавлять в него новые возможности.
* Кроссплатформенность. Visual Studio Code есть для всех популярных операционных систем, так что с ним можно работать вне зависимости от ОС, которую вы предпочитаете. К тому же это позволяет быстрее и удобнее переносить проекты с одного устройства на другое.
* Поддержка разных языков. Есть инструменты, которые предназначены специально для какого-то языка, например Python или C++. Visual Studio Code поддерживает огромное количество языков программирования, и они легко подключаются. Поэтому с его помощью можно вести проекты, написанные на нескольких языках одновременно.
* Гибкость. У VS Code множество расширений и удобных функций. Помимо вещей, стандартных для редактора кода, вроде подсветки синтаксиса, он «умеет» много чего еще. Его можно легко и гибко настроить под себя в зависимости от потребностей разработчика и технологий, с которыми он работает.

Недостатки Visual Studio Code

* Ограниченная функциональность. Конечно, это зависит от того, с чем сравнивать VS Code. По сравнению с рядом других редакторов он, наоборот, более функционален. Но если сравнивать с IDE — любой редактор проигрывает, так как возможности у него намного ниже.
* А для некоторых языков возможности IDE критичны. Поэтому работать в редакторе кода для разработчиков на этих языках будет не очень удобно. Например, в VS Code есть модули для [Java](https://blog.skillfactory.ru/glossary/java/" \t "_blank), но Java-разработчики все же предпочитают пользоваться специализированными средами. У них больше возможностей по работе со специфическими особенностями языка вроде перевода программы в байт-код.
* Медленная работа. На старых компьютерах редактор может долго запускаться и работать медленнее. Особенно это характерно для устройств, где установлен жесткий диск HDD, а не твердотельный накопитель SSD, или мало оперативной памяти.

Так происходит из-за того, что VS Code написан на Electron — это фреймворк, который позволяет писать программы для ПК с помощью HTML/CSS и JavaScript. Внутри фреймворка – целый браузер, отвечающий за отображение HTML и CSS. Соответственно, написанные на Electron программы такие же ресурсоемкие, как обычные браузеры, а значит, на слабых устройствах могут работать медленно. Это справедливо и для VS Code.

У языков программирования есть заданный набор команд и определенный синтаксис: то, как должны записываться команды.

Язык гипертекстовой разметки сайта, или HTML (HyperText Markup Language), - это код, помогающий структурировать содержание каждой веб-страницы. С помощью HTML разработчик собирает «скелет». После чего работает с CSS-кодом для стилизации страницы - покрывает «кожей». В финале запускает «кровеносную, пищеварительную и дыхательную системы», программируя на JavaScript.

Что такое язык HTML:

* HyperText (гипертекст) - текст, соединяющий связанные между собой элементы.
* Markup (разметка) - руководство по стилю набора текста.
* Language (язык) - код, который компьютерная система понимает и использует для интерпретации команд.

HTML-документ создаётся в обычном текстовом редакторе - чаще всего в «Блокноте». Есть и специализированные приложения вроде Notepad++ или Visual Studio Code - с подсветкой кода.

С помощью языка разметки HTML браузер делает запрос по адресу, который ввёл пользователь, и получает файл в формате «.html». Браузер распознаёт код, выбирает знакомые для себя сигналы: понимает, что написать словами, где поставить заголовок и какой именно. Таким образом код из файла преобразуется в необходимые визуальные объекты.

Язык HTML помогает структурировать информацию. Это могут быть команды добавить картинку, разбить текст на абзацы, добавить заголовки и подзаголовки, создать список, вставить таблицу, нарисовать какой-либо элемент. Вёрстка страницы будет выглядеть так, как её закодирует разработчик.   
Что представляет собой HTML: набор команд, который после обработки превращается в визуальное представление. Команда - это тег, состоящий из имени, расположенного между символами «меньше» и «больше» (<h1>). Есть парные теги. Правила вложенности у каждого свои. Например, вот так выглядит строка в списке:

<ul>

<li>Название элемента в списке</li>

</ul>

Из-за незакрытого или неправильно закрытого тега вёрстка может ломаться.   
Простая HTML-страница состоит из трёх тегов: <html>, <head> и <body>. В документе <head> и <body>используют только раз.

* <html> - идёт в документе сразу после «доктайпа» — обозначения типа документа. По нему браузер определяет версию HTML и понимает, как правильно отобразить страницу.
* <head> хранит важную служебную информацию - заголовок страницы и кодировку.
* <body> - хранит содержимое страницы. Именно так код отображается в браузере. Тексты и картинки добавляются внутрь этого тега.

Чтобы специалист понимал, потребуется ли ему HTML в работе и готов ли он его изучать, нужно взвесить все плюсы и минусы этого языка.

Достоинства языка:

* Подходит почти для всех браузеров, а код можно написать в любом текстовом редакторе.
* HTML – можно использовать бесплатно.
* Чтобы освоить азы языка разметки гипертекста, достаточно пары часов. У него простой и понятный синтаксис.
* Подходит для профессиональных разработчиков сайтов и для любителей.
* Статические сайты на HTML, которым не нужно обращаться к базам данных, быстро загружаются, потому что меньше весят и в них отсутствует «мусорный код». Они дешевле в разработке, и для таких сайтов подойдёт самый дешёвый хостинг.
* HTML – интегрируется с другими языками программирования.
* HTML –элементы содержат в себе семантику и работают с учётом цифровой доступности – ими уже могут пользоваться незрячие люди.
* HTML – способен сам закрывать теги и пропускать синтаксические ошибки. В отличие от ошибки в JS, страница не упадёт, а всё равно загрузится, пусть и с непредсказуемыми искажениями.

Минусы языка:

* Все повторяющиеся изменения на сайте придётся вносить вручную. Например, заменить на каждой странице пункты меню и навигации. То же самое с созданием новых страниц — даже если структура повторяется, нужно создавать каждую страницу отдельно.
* Чтобы сделать хороший сайт, который будет быстро открываться на любом устройстве, в режиме чтения или со скринридера, недостаточно везде использовать < div>. Существует более 100 HTML-элементов, нужно знать их семантику, где именно их применять. Например, не всегда то, что на макете выглядит таблицей, ей является. Для этого нужно читать [спецификацию по HTML](https://html.spec.whatwg.org/multipage/) и уметь искать в ней нужную информацию.
* Если нужно, чтобы сайт выглядел красиво и реагировал на действия пользователя, придётся учить ещё CSS и JS.

PHP – это скриптовый язык программирования. Он имеет открытый исходный код. Изначально создавался для разработки веб-приложений, но в процессе обновлений стал языком общего назначения.

PHP создавался для веб-разработки, и для этого он в основном и используется. На нём часто пишут динамические страницы и небольшие веб-программы. Вот несколько примеров его использования:

* работа с базами данных;
* создание системы комментариев;
* разработка форумов;
* разработка поиска по сайту;
* создание страницы для регистрации и входа в систему.

Начнём с особенностей. У PHP**слабая динамическая типизация:** это значит, что типы переменных определяются во время выполнения программы, разные типы можно использовать вместе, а неявные преобразования выполняются автоматически.

Такая типизация — **это и плюс, и минус.** С одной стороны, код проще, гибче, он лучше читается и его легче писать, но, с другой стороны, вероятность ошибок выше, а найдёте вы их уже после запуска программы.

Плюсы PHP:

* PHP -бесплатный, а его код лежит в открытом доступе.
* Для PHP - создано множество библиотек, баз данных и фреймворков.
* PHP - совместим почти со всеми современными серверами.
* PHP - гибкий и даёт много свободы.
* PHP -хорош для новичков, ему легко научиться.
* Простая иэлегантная система классов и объектов.
* Простой синтаксис.

Минусы PHP:

* Чтобы работать с PHP, нужно знать как минимум**HTML,** а лучше ещё и **CSS.** Знание **JavaScript** тоже пригодится.
* У простоты и гибкости PHP тоже есть обратная сторона: написать плохой код очень просто, ошибиться легко, а найти ошибку тяжело. Из-за этого поддерживать код бывает непросто.
* **Безопасность.** Во-первых, проблему создаёт то, что код лежит в открытом доступе: если кто-то найдёт в коде баг, он может использовать это в своих целях. Во-вторых, из-за простоты языка и большого количества разработчиков код часто низкого качества и в нём не предусмотрена защита.
* **Скорость.** PHP не самый производительный язык: например, его конкурент [Javascript — быстрее](https://benchmarksgame-team.pages.debian.net/benchmarksgame/fastest/php-node.html" \t "_blank).
* Нет четкой **системы в названиях** функций стандартной библиотеки: в некоторых есть сокращения, в некоторых нет *(*call\_user\_funcvs*.*create\_function*)*. В некоторых есть подчёркивание, в некоторых нет *(*issetvs*.*is\_null*)*, а, например, в названиях функций для работы со строками иногда бывает обозначение str, а иногда его не бывает. Такая **нестабильность усложняет работу.**

JavaScript — это язык программирования, который используют для написания frontend- и backend - частей сайтов, а также мобильных приложений. Часто в текстах и обучающих материалах название языка сокращают до JS. Это язык программирования высокого уровня, то есть код на нем понятный и хорошо читается.

[Frontend](https://skillfactory.ru/frontend-razrabotchik/?utm_source=blog%20&utm_medium=referral&utm_campaign=chem-otlichaetsya-backend-i-frontend&utm_content=coding_fr&utm_term=text) — это клиентская часть продукта (интерфейс, с которым взаимодействует пользователь). В случае с сайтом, её формирует и выводит на экран браузер, который работает с [HTML](https://blog.skillfactory.ru/glossary/html/), CSS и JavaScript.

[Backend](https://skillfactory.ru/backend-razrabotchik-na-golang/?utm_source=blog%20&utm_medium=referral&utm_campaign=chem-otlichaetsya-backend-i-frontend&utm_content=coding_go&utm_term=text) — это внутренняя часть продукта, которая находится на сервере и скрыта от пользователей. Для ее разработки могут использоваться самые разные языки, например, Python, PHP, Go, JavaScript, Java, С#.

JavaScript - поддерживают все популярные браузеры. Во frontend-части сайтов язык используют для создания интерактива (анимаций, всплывающих форм, автозаполнения), так как он связан с [HTML](https://blog.skillfactory.ru/glossary/html/) и [CSS](https://blog.skillfactory.ru/glossary/css/) и может ими манипулировать. В backend-части с языком JavaScript работают на платформе Node.js. С ее помощью, например, разрабатывают серверные веб-приложения и подключают библиотеки.

JavaScript - называют языком скриптов или сценариев.

Скрипты — это набор инструкций, которые выполняются при загрузке страницы. Браузер самостоятельно интерпретирует код на JavaScript, для этого даже не требуется компиляция (перевод языка программирования в машинный код).

Скрипты можно прописать внутри кода страницы или подключить к HTML отдельным файлом.

JS — мультипарадигменный язык, так как он поддерживает разные парадигмы (стили) программирования:

* Объектно-ориентированный — представляет программу как систему отношений между объектами — наборами данных и функций.
* Императивный — в этом стиле программирования используют инструкции, которые выполняются последовательно, а для доступа к данным используются переменные, значение которых может быть переписано в ходе вычисления.
* Функциональный — в нем вычисление основано на математических функциях. Так как функциональное программирование не опирается на переменные, которые могут менять значение, результаты запросов будут всегда одинаковыми.

Веб-страницы частично обрабатываются с помощью JavaScript на компьютере пользователя. Это снижает нагрузку на сервер: часть операций выполняется без запросов к нему, что экономит время и трафик.

В веб-приложениях HTML отвечает за разметку страницы. Расположение контента, внешний вид интерфейса описывается с помощью CSS, а JavaScript отвечает за интерактивность и функционал этого интерфейса. Вот примеры задач, которые решает JS:

* Взаимодействие с пользователем. Всплывающие окна с полем для ввода почты или других данных, уведомления об использовании cookie.
* Обработка данных HTML. JS помогает заполнить необходимые поля в формах, проверять буквенные и числовые значения, а также управлять HTML - элементами на странице. Например, в конструкторе лендингов [Tilda](https://tilda.cc/ru/" \t "_blank) можно перетаскивать блоки и по клику менять шрифты, цвета текста, фонов и другие значения.
* Анимация. Это не только движущиеся объекты на странице и кнопки, но и карусель в галерее или плавная прокрутка страницы. Например, [Pinterest](https://www.pinterest.ru/" \t "_blank) с помощью JavaScript создал эффект бесконечной ленты.
* Математические вычисления. С помощью JS-скриптов на сайтах реализованы калькуляторы, например подсчет суммы заказа в корзине интернет-магазина.

Как работает JavaScript в браузере:

Действия пользователя на странице вызывают события: клик на кнопке запускает анимацию, захват мышкой и перемещение курсора двигают объект по странице. Каждое из этих действий выполняется с помощью скриптов, написанных на JavaScript. У них есть определенный алгоритм работы:

1. Пользователь совершает действие - нажимает на кнопку «Зарегистрироваться».
2. Браузер фиксирует выполнение определенных условий - клик мыши в конкретной области на странице. За реакцию на действие отвечает команда onclick, которая запускает прописанный JS-код.
3. Затем запускается JS-код, который прописан для этих условий, - анимация нажатия кнопки и открытие формы регистрации.
4. Страница полностью или частично обновляется - в зависимости от настроек форма может открыться в этой же или в соседней вкладке.

К преимуществам JavaScript можно отнести следующие моменты:

1. Незаменимость в веб-разработке. JS – это основная «технология» для клиент-серверных моделей и программирования «для интернета».
2. Наличие полной интеграции с версткой страниц и серверной частью.
3. Рациональность применения и простоту. Решить элементарную задачу при помощи JS удается за несколько минут. Сам ЯП обладает понятным синтаксисом, который не требует углубленного изучения.
4. Скорость работы. JS может сделать итоговое приложение более быстрым и удобным.
5. Производительность.
6. Комфортность использования пользовательских интерфейсов.
7. Наличие собственной мощной экосистемы (инфраструктуры). Особо заметно в последние годы. Пример – появление огромного количества полезных фреймворков, которые подойдут для любого случая.

### Минусы:

Почему нужно использовать JavaScript, понятно – это удобный и функциональный язык. Но у него есть свои недостатки. К ним относят:

1. Отсутствие возможности чтения и загрузки документов. Основная причина наличия этого минуса – соображения безопасности.
2. Отсутствие удаленного доступа. Полноценно для сетевого ПО соответствующий язык не используется.
3. Нестрогую типизацию.
4. Вольную трактовку типов. Нестыковки в коде будут игнорироваться.
5. Доступность для конкурентов. Связано это с высокой читаемостью исходного кода JavaScript.

Несмотря на соответствующие ограничения, языки программирования, все равно стремительно развивается. В последние годы в Сети появилось множество библиотек и инструментов на JavaScript. Поэтому обратить внимание на него при попытках написания веб-страниц рекомендуется каждому.

CSS — это язык описания внешнего вида документа, то есть он отвечает за то, как выглядят веб-страницы: цвет фона и декоративных элементов, размер и стиль шрифтов. Термин расшифровывается как Cascading Style Sheets (каскадные таблицы стилей). CSS взаимодействует с другим языком разметки — HTML, который отвечает за размещение элементов на странице.

Допустим, с помощью HTML текст уже расположен в верхней части документа. Задать для него цвет, тип шрифта и его кегль, сделать текст жирным или выделить курсивом можно и без использования CSS. Для этого в исходный код нужно добавить тег, который определяет начертание текста.

Как устроены таблицы стилей:

CSS, как и любой язык, имеет свой синтаксис. В нем есть правила - значения, которые определяют внешний вид элементов. CSS-правило состоит из селектора, CSS-свойств и их значений:

* Селекторы - это метки, которые помогают браузеру понять, к какой части HTML-кода нужно применить заданные параметры.
* CSS-свойства - это определенные параметры оформления, например цвет элемента или текста (color) или цвет фона (background).
* Значение - это просто значение, оно выражается текстом или числом, например черный (black).

Таблицы называются каскадными, потому что работают по принципу каскада - то есть правило, прописанное ниже, считается приоритетным.

Подключение стилей CSS:

Чтобы использовать CSS совместно с HTML, можно выбрать один из способов:

Весь код, написанный на CSS, прописывается в отдельном внешнем файле с расширением .css. Его подключают к HTML-странице при помощи тега <link href> — это служебный тег, который на странице не будет видно;

Прописать стили CSS внутри конкретного тега с помощью атрибута style;

Преимущества CSS:

* CSS играет важную роль, используя CSS, вы просто должны указать повторяющийся стиль для элемента один раз и использовать его несколько раз, поскольку CSS автоматически применит требуемые стили.
* Главное преимущество CSS заключается в том, что стиль последовательно применяется на множестве сайтов. Одна инструкция может управлять несколькими областями, что является преимуществом.
* Веб-дизайнерам необходимо использовать несколько строк программирования для каждой страницы, что повышает скорость работы сайта.
* Каскадная таблица не только упрощает разработку веб-сайта, но и упрощает обслуживание, поскольку изменение одной строки кода влияет на весь веб-сайт и время обслуживания.
* Он менее сложный, поэтому затраты на него значительно снижены.
* Это помогает формировать спонтанные и последовательные изменения.
* Изменения CSS удобны для устройств. Поскольку люди используют различные интеллектуальные устройства для доступа к веб-сайтам через Интернет, существует потребность в адаптивном веб-дизайне.
* Он обладает способностью изменять позиционирование. Это помогает нам определять изменения в расположении веб-элементов, которые присутствуют на странице.
* Эта экономия полосы пропускания заключается в значительном количестве незначительных тегов, которые неотличимы от беспорядка страниц.
* Пользователю легко настраивать онлайн-страницу
* Это уменьшает размер передаваемого файла.
* Недостатки CSS:
* CSS, начиная с CSS 1 и заканчивая CSS3, приводят к созданию путаницы среди веб-браузеров.
* С помощью CSS то, что работает с одним браузером, не всегда может работать с другим. Веб-разработчикам необходимо протестировать совместимость, запустив программу в нескольких браузерах.
* Существует дефицит безопасности.
* После внесения изменений нам необходимо подтвердить совместимость, если они появятся. Подобное изменение затрагивает все браузеры.
* Мир языков программирования сложен для не-разработчиков и новичков. Различные уровни CSS, то есть CSS, CSS 2, CSS 3, часто довольно запутанны.
* Совместимость с браузером (некоторые таблицы стилей поддерживаются, а некоторые нет).
* CSS работает по-разному в разных браузерах. IE и Opera поддерживают CSS как разную логику.
* При использовании CSS могут возникать межбраузерные проблемы.
* Существует несколько уровней, которые создают путаницу для не-разработчиков и новичков.

2.2 Основные принципы разработки сайта

Для начала разберёмся, что такое сайт.

Сайт — это место в интернете (от англ. site – место, позиция), где по одному адресу находятся одна или несколько веб-страниц.

Образно сайт можно сравнить с книгой, в которой под одной обложкой собрано несколько страниц. Так же, как и книги, сайты могут быть самого разного назначения:

* Развлекательные;
* Научные;
* Обучающие;
* Каталоги;
* Справочники и т.д.

В отличие от книг, сайты могут содержать не только тексты и картинки, но и музыку, видео, скрипты, игры, магазины, чаты, форумы, социальные сети и т.д.

Что такое верстка сайта?

Вёрстка определяет, как части сайта будут выглядеть в разных браузерах и на  разных устройствах.

Большая часть пользователей заходят в интернет с мобильных устройств — смартфонов и планшетов. Поэтому сайты должны иметь адаптивную вёрстку, чтобы их удобно было смотреть и на больших мониторах, и на маленьких смартфонах.

Сайты с устаревшей вёрсткой выглядят на смартфонах либо чересчур мелко, либо не помещаются на маленьком экране и их приходится скролить как по горизонтали, так и по вертикали, что очень неудобно.

Сайты с современной вёрсткой сами определяют, с какого устройства их смотрят, и отображаются так, чтобы пользователям было удобно.

Что такое домен сайта (доменное имя).

Домен сайта — это его адрес в сети интернет. Образно его можно сравнить с адресом квартиры, который состоит из: названия страны, города, улицы, номера дома и квартиры.

Доменное имя также многоуровневое и содержит домен первого уровня, обозначающий принадлежность сайта к той или иной стране или назначение сайта.

Что такое хостинг сайта.

Сайт должен быть доступен в сети интернет круглосуточно и открываться за доли секунды. Поэтому его невозможно держать на домашнем компьютере — его мощности и скорости интернет-соединения будет недостаточно. Поэтому сайты «живут» на хостинге.

Хостинг — это специализированная услуга, предоставляющая мощные серверы (высокопроизводительные компьютеры с дисками высокой ёмкости), соединённые с интернетом высокоскоростными линиями связи.

Различают хостинги по тому, сколько ресурсов они предоставляют сайту:

* Обычный хостинг - один сервер могут делить тысячи или десятки тысяч сайтов. Это вызывает неудобства, потому что некоторые сайты могут создавать такую большую нагрузку на сервер, что он не справляется и все остальные сайты «тормозят» или вообще не открываются. Вполне подходит для новичков.
* VDS-хостинг - это аренда виртуального сервера, когда каждому сайту или сайтам одного пользователя выделяются определённые ресурсы, гарантирующие его бесперебойную работу. Хороший вариант для доходных сайтов, магазинов и большинства других сайтов.
* Выделенные серверы - каждому пользователю выделяется сервер, который он может использовать «единолично» и настраивать под свои задачи. Это нужно для высоконагруженных интернет-магазинов, порталов с высокой посещаемостью, игровых и MMO сайтов.

Что такое индексация сайта.

1. Найти сайт в интернете можно тремя способами:
2. Набрать в адресной строке браузера его точный адрес;
3. Перейти по ссылке с другого сайта;
4. Ввести запрос в поисковую систему.

Поисковики (Яндекс, Гугл, Бинг и др.) получают ежедневно миллионы запросов и должны находить нужную пользователям информацию за доли секунды.

Они не могут по каждому запросу сканировать весь интернет, в котором миллиарды интернет-страниц — на это не хватит никаких ресурсов и по времени это очень долго.

Поэтому поисковые системы создают текстовые копии всех известных интернет-страниц. База этих копий называется index, а поиск и создание копий страниц — индексирование.

Индексирование — процесс постоянный, так как сайты растут, появляются всё новые страницы, изменяется содержимое старых, создаются новые сайты. Поэтому специальные поисковые боты периодически обходят известные им ресурсы, находят и индексируют новые и изменённые страницы. А также обнаруживают ссылки на неизвестные им сайты и индексируют их.

Что такое оптимизация сайта?

Это подготовка сайта для более эффективной индексации поисковыми системами. Хорошо оптимизированный сайт пользователи видят на первых строчках в результатах поиска и чаще переходят на него.

Давайте разберем и объясним основные принципы создания различных сайтов:

1. Интернет - глобальная информационная сеть, части которой логически связаны друг с другом единым адресным пространством на основе протокола TCP/IP. Интернет состоит из множества взаимосвязанных компьютерных сетей и обеспечивает удаленный доступ к компьютерам, электронной почте, доскам объявлений, базам данных и дискуссионным группам.

Сегодня мы можем использовать сеть для получения видеоизображений из любой точки мира, отправки электронных писем с фотографиями и тому подобное.

1. Веб-дизайн - это процесс создания визуальной оболочки сайта, его структуры и навигационной системы в форме макета.

Дизайн веб-сайтов и их отдельных элементов, создание графических рекламных материалов в Интернете. Обычно услуги веб-дизайна предлагаются студиями веб-дизайна.

1. Веб-сервер - сервер, предоставляющий информацию в сервисе глобального соединения. Веб-сервер хранит и делает доступными для внешней сети данные, организованные как веб-страницы. Веб-сервер отвечает за обработку запросов клиентов к сайту и выполнение (CGI, JSP, ASP, PHP и других приложений).
2. Веб-страница- самостоятельная часть сайта; документ с уникальным адресом (URL). Сайт может быть статическим или динамическим. Обычно веб-страницы организованы в виде гипертекста, содержащего текст, графику, звук, видео или анимацию. В Интернете вы можете просматривать веб-страницы с помощью браузера. Безусловно, возможно создание Web-страницы и без использования графики при помощи шрифтов, скриптов и таблиц стилей, и это будет красиво и стильно. Но ведь окончательный вид документа зависит от большого числа различных факторов, таких как: ширина окна браузера, предварительные настройки браузера, принятые по умолчанию размер шрифта, его имя и цвет. К тому же не все скрипты и стили поддерживаются всеми браузерами. Если же будет использована графика, то посетитель страницы увидит ее точно такой, какой ее создали.
3. Главная страница - это стартовая страница сайта. Обычно главная страница несет основную презентацию и навигационную нагрузку.
4. Личные страницы - веб-страницы, принадлежащие физическим лицам. Содержание и дизайн личной страницы зависит исключительно от ее автора.
5. Веб-сайт - это коллекция веб-страниц с повторяющимся дизайном объединенных по смыслу, навигации и физическому расположению на одном веб-сервере. Как правило, они посвящены одной тематике.

На каждой Web- странице есть, так называемая «Главная/Домашняя» страница. Все пользователи могут попасть на неё, нажав на определённую гиперссылку.

Гиперссылка **–** это текст или изображение на веб-странице при нажатии на неё, которое позволяет пользователю перейти на другую веб-страницу или веб-сайт. Многие из разработчиков считают, что быть хорошим дизайнером - это врожденная способность, что творческий подход - это то, с чем вы сразу родились. Но самом деле, это далеко не так. Дизайн - это навык, который можно освоить, как и любой другой, если только захотеть.

На сайтах нередко присутствуют битые гиперссылки, которые ведут на несуществующие файлы и требуют удаления. Когда на одном изображении размещается несколько гиперссылок, направляющих на разные страницы или действия, такой объект называют гиперкартой. На ней размещают подсказки с указанием, куда ведет каждая область.

Глава 3. Разработка сайта-визитки для фотостудии «Фото+» г. Королёв

3.1 Правила создания сайта

При разработке сайта, мы отталкиваемся, от общих правил его создания.

* 1. Содержание. Почему пользователи приходят на сайт? Им нужна информация. [Содержание сайта](https://sansans.ru/blog/seo-smo-smm/7-sposobov-vyzvat-u-posetitelej-doverie-k-sajtu.html), его текстовое наполнение является важнейшим критерием качества. Чем полезнее и актуальнее информация, опубликованная на сайте, тем выше интерес пользователей и больше посещаемость. Контент должен отвечать тематике ресурса и нести практическую пользу. Содержание – основа сайта, без которой невозможно представить существование хорошего веб-проекта.
  2. Внешнее оформление. Не менее важный критерий – это оформление. Как и человека, сайт встречают по одежке – его дизайну. Красивый и удобный ресурс сразу располагает к себе, вызывает симпатию у посетителей. Непонятный и безобразно оформленный сайт хочется поскорее закрыть. Пользователи начинают оценивать сайт с внешнего вида и лишь после этого переходят к изучению содержимого. По [дизайну веб-ресурса](https://sansans.ru/blog/web-dizajn/sovremennye-instrumenty-dlya-veb-dizajna.html) будут судить о компании в целом, о ее качестве и надежности. Помимо аккуратности при разработке дизайна надо руководствоваться принципами актуальности, уникальности, соответствия тематике, удобства для пользования. Важно учитывать технические тонкости оформления, а также эстетичность.
  3. Система навигации и структура. [Хороший сайт](https://sansans.ru/blog/web-dizajn/kakie-znaniya-i-umeniya-vy-poluchite-posle-prokhozhdeniya-kursa-web-dizajn.html) должен быть тщательно продуман с точки зрения структуры и удобства нахождения информации. Все материалы должны быть упорядочены по разделам и категориям. Четкая структура не только упрощает поиск информации пользователями, но и способствует более быстрой и правильной индексации сайта роботами.

Качественная система навигации обеспечивает посетителям легкое и удобное перемещение по страницам. На сайте не должно быть пустых страниц, на которых пользователи увидят текст «страница находится в разработке». Все это портит впечатление о веб-проекте, снижая доверие к компании.

* 1. Стилистика и грамматика. [Контент на сайте](https://sansans.ru/blog/seo-smo-smm/kak-prodvigat-svoj-sajt-na-osnove-kontenta.html) должен иметь хорошую стилистику. Наличие в текстах ошибок считается недопустимым для хорошего сайта. Также необходимо обеспечить правильную верстку, подобрать удобный для чтения шрифт, выбрать цвет и размер. Все это влияет на отношение пользователей к сайту и поведенческие факторы, от которых зависит [успех поискового продвижения](https://sansans.ru/blog/seo-smo-smm/kursy-seo-optimizatsii-i-prodvizheniya-sajta-chemu-vy-nauchites.html).
  2. Качество программирования. На сайте нужен [грамотный программный код](https://sansans.ru/blog/programmirovanie/chto-vy-budete-znat-i-umet-posle-prokhozhdeniya-kursa-po-php-mysql.html), а все программные решения должны работать без сбоев. Любой пользователь должен получить возможность увидеть ваш сайт в качественном формате. Важно настроить правильное отображение ресурса в различных браузерах и на разных устройствах.
  3. Обновление содержимого. Хороший веб-ресурс должен быть «живым». Необходимо поддерживать активность. Для этого следует ежедневно добавлять новые материалы, отвечать на комментарии и вопросы посетителей, обновлять имеющуюся информацию, делая ее актуальной. Нельзя забывать про сайт и оставлять его без внимания.
  4. Правильная оптимизация и реклама. Пользователи находят сайты через поисковые системы, баннеры на других ресурсах, в каталогах и социальных сетях. Чтобы люди нашли вас, необходимо [оптимизировать веб-ресурс](https://sansans.ru/blog/seo-smo-smm/optimizatsiya-sajtov-dlya-poiskovykh-sistem.html). Для этого подбираются поисковые запросы, которые будут приводить на сайт целевую аудиторию. В качестве инструментов рекламы можно использовать контекст, баннеры, сообщения в соцсетях, ссылки на других тематических проектах.
  5. Интерактивность. Оперативное взаимодействие с пользователями – еще одна составляющая хорошего сайта. На каждый вопрос необходимо давать своевременный ответ. Если на сайте используется система «онлайн-консультант», то необходимо обеспечить наличие специалиста на месте. Для улучшения интерактивности можно использовать дополнительные сервисы и программные решения.
  6. Уникальность контента и цитирование. На сайте должны использоваться [уникальные материалы](https://sansans.ru/blog/seo-smo-smm/kak-pravilno-napisat-seo-statyu-sovet-pervyj.html). Если размещаются чужие тексты и фотографии, следует указывать авторство и ссылку на источник. Поисковые системы не одобряют размещение копипаста, поэтому его наличие может привести к наложению санкций.
  7. Ограничение рекламы. Откажитесь от большого количества рекламных объявлений на своем сайте. Они создают негативное отношение к ресурсу со стороны посетителей. Люди приходят на ваш сайт для получения ценной информации. Если в этом им помешает реклама, они покинут площадку. Обилие рекламы отрицательно влияет и на поисковое продвижение.

Соблюдайте правила хорошего сайта, ведь они не такие сложные. Следуя им, вы повысите популярность своего веб-проекта и сформируете лояльную аудиторию посетителей, которые в дальнейшем смогут стать вашими клиентами и подписчиками.

3.2 Стили

Теперь поговорим о стилях, которые применяются для красоты и удобства на разработанных сайтах.

Стиль – это основное направление в оформлении сайта, которое прослеживается в большинстве элементов страницы. Говоря о стиле сайта, мы подразумеваем, что все элементы связаны друг с другом и обеспечивают целостное восприятие страницы.

CSS (Cascading Style Sheets) – язык таблиц стилей, который позволяет прикреплять стиль (например, шрифты и цвет) к структурированным документам (например, документам HTML и приложениям XML).

Обычно CSS-стили используются для создания и изменения стиля элементов веб-страниц и пользовательских интерфейсов, написанных на языках HTML и XHTML, но также могут быть применены к любому виду XML-документа, в том числе XML, SVG и XUL. Отделяя стиль представление документов от содержимого документов, CSS упрощает создание веб-страниц и обслуживание сайтов.

К отдельным составляющим стилистики как правило относят:

* цветовую гамму;
* [типографику](https://idbi.ru/blogs/blog/tipografika-v-veb-dizayne);
* расположение элементов;
* стиль изображений и другого графического контента и прочее.

Все стили я писала в Tailwind.

Tailwind -  это CSS-фреймворк, предлагающий обширный каталог классов и инструментов для облегчения стилизации сайта или приложения. Основная идея состоит в том, что при разработке проекта вам не нужно иметь дело с таблицами стилей и беспокоиться о том, как переопределить тот каскад из 10 селекторов. Одна вещь, которую большинство разработчиков могут найти непривлекательной в Tailwind CSS, это то, что ваша разметка выглядит намного более загруженной, чем вы могли бы желать. Taildwind предоставляет большое разнообразие CSS-классов, каждый из которых имеет собственную функцию. Вместо традиционного создания класса .btn путём добавления множества атрибутов в Tailwind можно просто применить следующие классы bg-blue-500 py-2 px-4 rounded к кнопке или создать класс .btn и вместе с утилитарным классом применить его к селектору. Tailwind настроена почти полностью на простом JavaScript. Для этого вам нужно сгенерировать конфигурационный файл Tailwind для проекта. Рекомендуется создать файл tailwind.js в корне проекта. Утилита CLI помогает легко с этим справиться.

Способы добавления CSS стилей.

Стандартная схема предполагает не один вариант использования стилей в табличках к веб-странице:

* Встроенный – подобным способом можно вставлять язык таблички стиля в начальный тег HTML-элементов;
* Внутренние таблицы со стилем – определяющие правила табличек, где используется стиль внутри тега style (чаще всего его можно найти в подразделе head);
* Внешние – они играют определяющую роль с правилами таблиц стиля в отдельных файлах, а потом подключают эти файлы к документам при помощи тега link.

На (Рисунке 6), приведён код для разработки стилей:

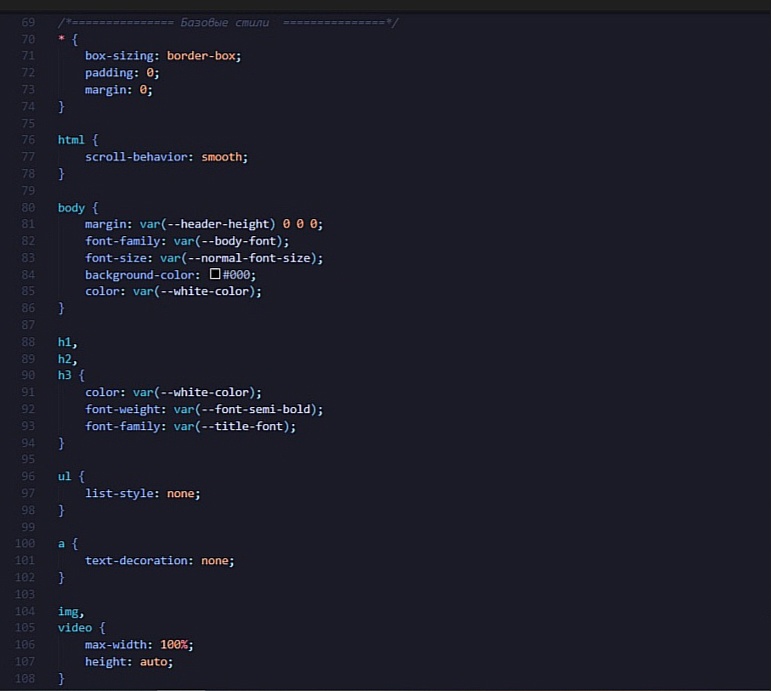


Рисунок 6. Код «Style» разработки сайта-визитки

HTML — это отправная точка. С нее начинаются все действия. Что необходимо предпринять? Дать понять HTML-документу, что мы бы хотели использовать определенные правила css. Нужно создать файл и сохранить его с расширением CSS. После того как файлы CSS и HTML сохранены, перезагружается страниц браузера. В это время заголовок светится цветом. Значит, команда была понята и применена. Все введено верно.

Конструктивные особенности «@import» тоже помогают добавлять документы CSS. Она помещается вверху html-документов. Но подобным методом лучше желательно не пользоваться, ведь нет гарантии, что по очереди загрузятся файлы css. А еще применение данного способа может быть чревато ошибками, возникающими во время загрузки веб-странички (Скрипты загружаются раньше, нежели подгружаются стили). По этой причине style css в html внутри тега рассматриваются в последний черед.

Тег <body> (от англ. Body – тело) предназначен для хранения содержимого веб-страницы (контента), отображаемого в окне браузера.

Информацию, которую следует выводить в документе, следует располагать именно внутри контейнера <body>. К такой информации относится текст, изображения, теги, скрипты JavaScript и т. Д.

<body> также применяется для определения цветов ссылок и текста на веб-странице. Подобная практика в HTML осуждается и взамен для указания цветовой схемы рекомендуется использовать стили, применяя их к селектору body.

Часто <body> используется для размещения обработчика событий, например, onload, которое выполняется после того, как документ завершил загрузку в текущее окно или фрейм.

Открывающий и закрывающий теги <body> на веб-странице не являются обязательными, однако хорошим стилем считается их использование, чтобы определить начало и конец HTML-документа.

HTML-элементы <h1>–<h6> представляют собой 6 уровней заголовков секций. <h1> это наибольший заголовок и <h6> - наименьший

HTML-элемент <ul> используется для неупорядоченного списка – в частности для маркированного списка.

Элемент [HTML](https://developer.mozilla.org/en-US/docs/Web/HTML) (или элемент привязки ) со [своим атрибутом](https://developer.mozilla.org/en-US/docs/Web/HTML/Element/a#href) создает гиперссылку на веб-страницы, файлы, адреса электронной почты, местоположения на той же странице или что-либо еще, к чему может обращаться URL-адрес.<a> [href](https://developer.mozilla.org/en-US/docs/Web/HTML/Element/a" \l "href)

Содержимое каждого из них <a> должно указывать на назначение ссылки. Если href – атрибут присутствует, нажатие клавиши ввода при фокусировке на <a>элементе активирует его.

3.3. Разработка сайта

Начнем описание сайта с того, что рассмотрим меню главной страницы. На ней находится разделы: Главная, Портфолио, Услуги, Контакты (Рисунок 7):



Рисунок 7. Разделы главного меню

Web-ресурс — это сложная структура со своими правилами и сложившимися традициями.

Хедер сайта — это голова, центр полетов, отвечающий за навигацию и юзабилити. Над хедером придется кропотливо поработать, ведь от каждой детали в этом блоке зависит, насколько людям будет удобно пользоваться вашим сайтом.

Header — так называется шапка сайта на английском языке. Это информационный блок в виде узкой полосы в верхней части каждой web-страницы.

Шапка — блок, который постоянно на виду, ведь большинство web-мастеров закрепляют его в верхней части страницы. У хедера три главных функции:

* идентификация — логотип и название компании в шапке сообщают пользователю, с чьим ресурсом он сейчас взаимодействует;
* навигация — ссылки в хедере помогают находить нужные разделы сайта и быстро перемещаться между ними;
* отстройка от конкурентов — среди однотипных вкладок пользователь предпочтет сайт с удобным и понятным хедером или, наоборот, с оригинальным и нестандартным.

В шапку выносится ключевая информация о вашем проекте. Здесь вы сообщаете пользователям, кто вы, чем занимаетесь, что предлагаете, как устроен ваш сайт, как с вами связаться. Но тут важно не терять голову и сосредоточиться на самом главном: напихав полную шапку информации и кнопок, вы рискуете отпугнуть пользователя. Перегруженный хедер — плохой хедер.

Информация в шапке должна быть хорошо структурирована и продумана. Оформление меню влияет на юзабилити — удобство пользования ресурсом. А юзабилити — это один из ключевых аспектов, влияющих на поведенческие факторы, важные для продвижения сайта в поисковых системах.

Проанализируйте примеры шапок сайтов, которые вам нравятся. У них много общего, но все же набор элементов отличается. Все зависит от ниши бизнеса, особенностей маркетинговой стратегии, самой структуры сайта и целей web-ресурса. Перечислим стандартные элементы хедера.

На (рисунке 8) представлено разработанное главное меню сайта.

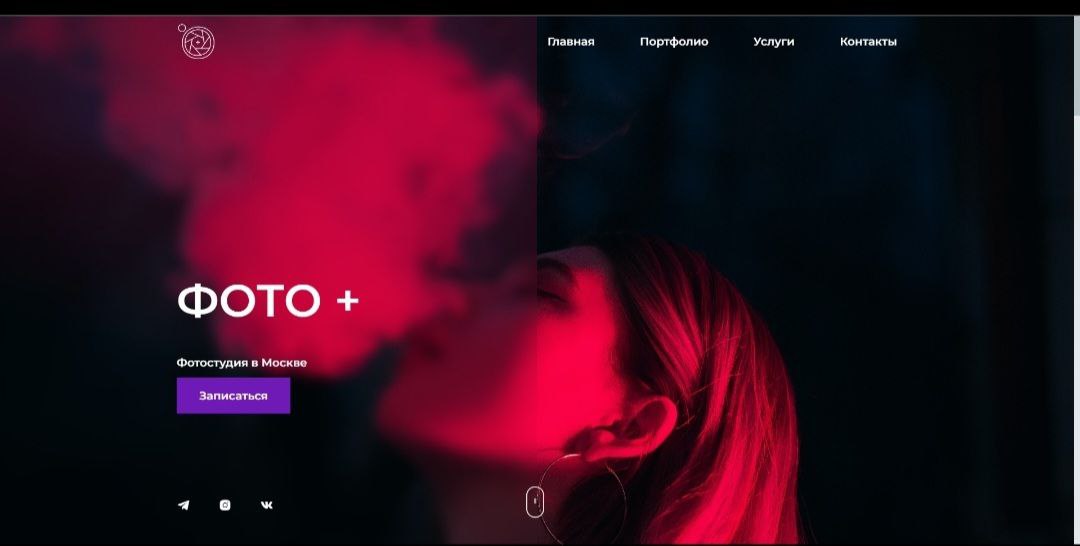


Рисунок 8. Разработанное меню и шапка сайта

Наличие в шапке веб-сайта кнопок для перехода в социальные сети неоднозначная история. С одной стороны, так продвигаются аккаунты в социальных сетях, но с другой - уводите пользователя на другие ресурсы, а это плохо для поведенческих факторов и конверсии.

Часто web-мастера просто дублируют посты из аккаунтов социальных сетей на главной, если это вписывается в специфику сайта, или ставят ссылки на социальные сети в футере.

Далее идёт разработка портфолио(Рисунок 9):

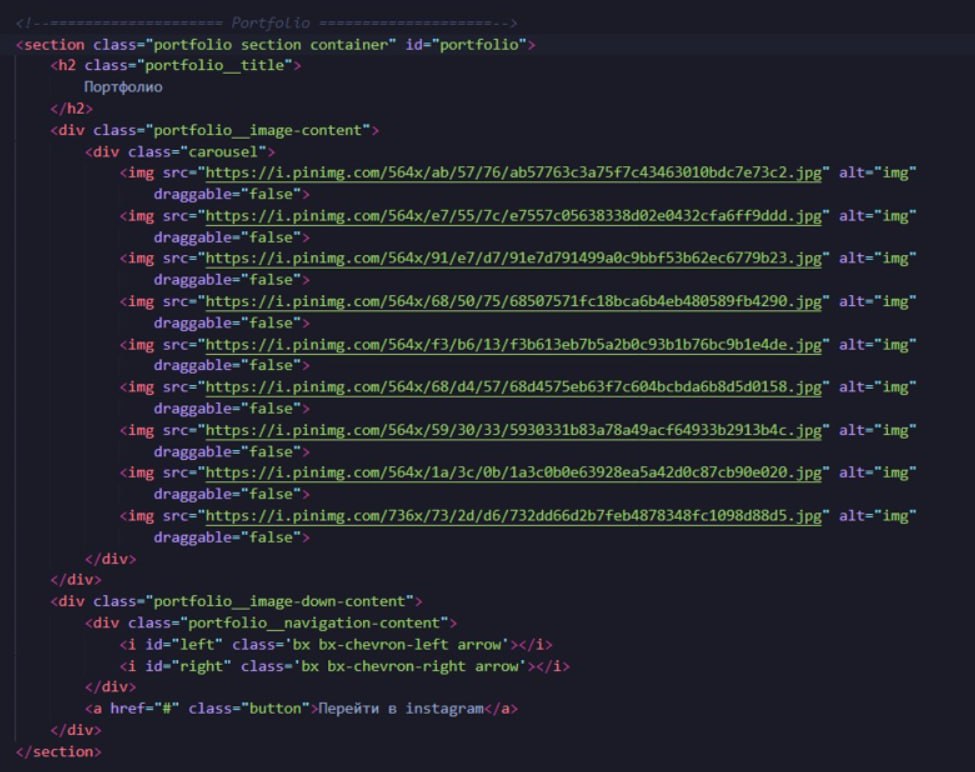


Рисунок 9. Разработанный код для портфолио

На (рисунке 10) представлено готовое портфолио сайта. На нем можно увидеть пример работ фотостудии.

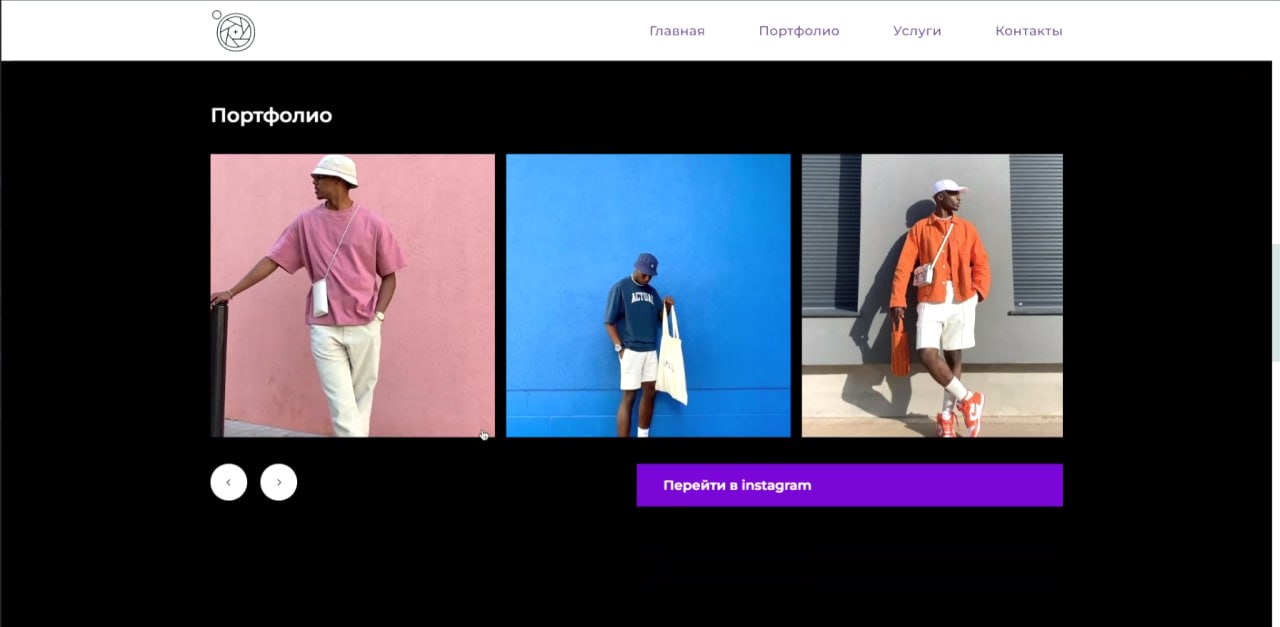


Рисунок 10. Готовое портфолио на сайте

Следующий раздел сайта – это раздел услуг. Ниже можно ознакомиться с представленным кодом услуг (Рисунок 11)

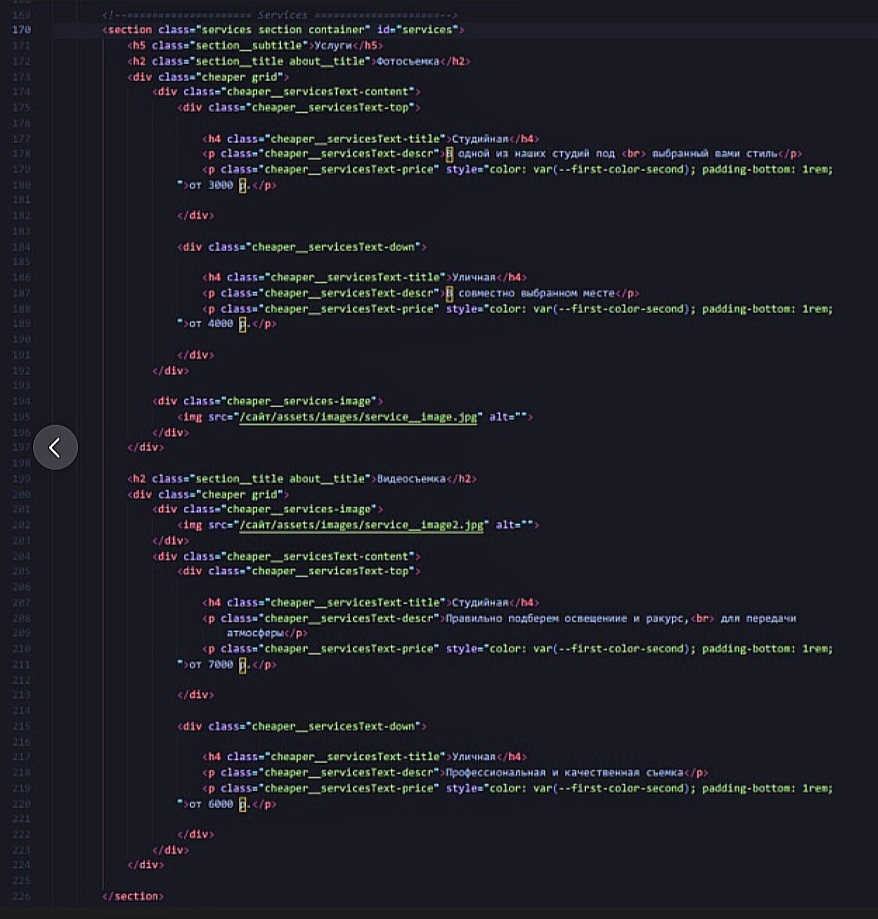
.

Рисунок 11. Код предоставляемых услуг

В этом разделе сайта, можно ознакомиться с предлагаемыми услугами фотостудии и выбрать подходящую заказчику на выбор услугу (Рисунок 12).

В представленные услуги входит:

* Два вида фотосъёмки: Студийная и уличная.

Студийная – В одной из наших студий под выбранный вами стиль (от 3000 тысяч рублей).

Уличная – В совместно выбранном месте (от 4000 тысяч рублей.)

* Два вида видеосъёмки: Студийная и уличная.

Студийная – Правильно подберем освещение и ракурс, для передачи атмосферы (от 7000 тысяч рублей).

Уличная – Профессиональная и качественная съемка (от 6000 тысяч рублей).

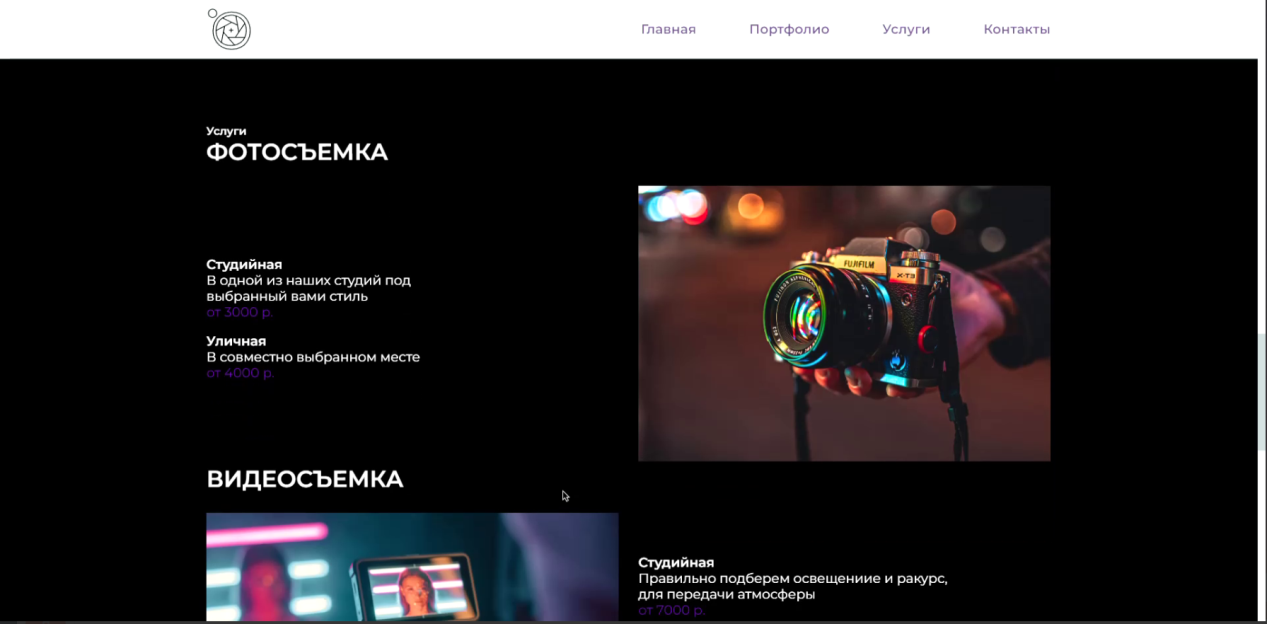


Рисунок 12. Предоставляемые услуги

И последняя разработка сайта, это – подвал (обратная связь в виде контактной информации). Здесь находятся контакты, адрес и почта. (Рисунок 13):



Рисунок 13. Код контактной информации сайта

Тег <footer> выделяет нижний колонтитул (“подвал”) веб-сайта или раздела. Как правило, в нем размещается информация об авторских правах, контактные данные, ссылки для навигации и т.д.

В пределах одной страницы можно разместить несколько тегов <footer>. К примеру, можно разместить нижний колонтитул внутри тега [<article>](https://ru.w3docs.com/uchebnik-html/html-teg-article.html) для хранения относящейся к статье информации (ссылки, сноски, и т.д.).

В нем могут содержаться другие HTML элементы, кроме тегов <footer> и [<header>](https://ru.w3docs.com/uchebnik-html/html-teg-header.html).

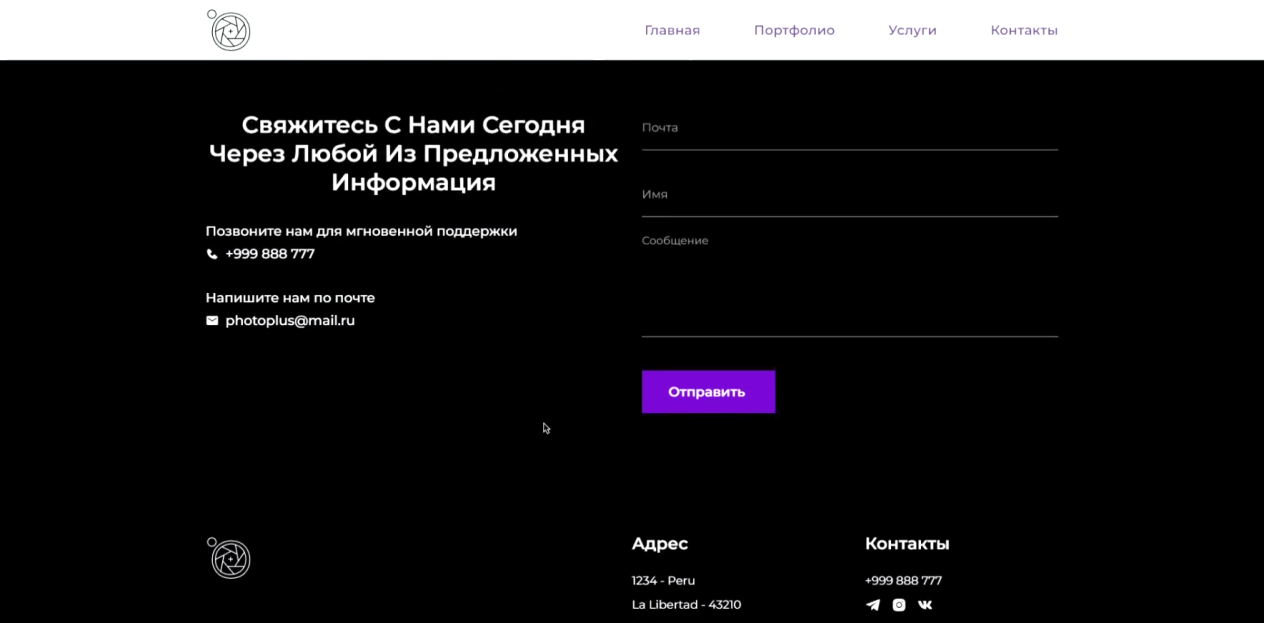


Рисунок 14. Контактная информация фотостудии

3.4. Основы техники безопасности сайта

Сейчас защита сайта от DDoS-атак, вирусов и взлома как никогда актуальна. Такие угрозы могут навредить сайту и пользователям, посещающим его. При игнорировании базовых принципов безопасности сайта, в сеть могут быть слиты пароли, данные платежных карт, клиентские базы, сайт может отображать нежелательный контент или рекламу, и даже использоваться для осуществления атак и заражения других сайтов.

Поисковые системы негативно относятся ко взломанным сайтам, применяя санкции, в результате которых теряется значительное количество трафика и позиции.

Чтобы сохранить работоспособность сайта и информацию, хранимую в нём, необходимо своевременно оградить его от атак, а для этого важно всесторонне защитить ресурс.

## Виды взлома:

Наиболее часто взломы происходят через веб-атаки сайтов из-за наличия в них разного рода уязвимостей. Далее рассмотрим частые варианты взломов сайтов.

* SQL-инъекции. Используются для извлечения, добавления, изменения и удаления данных в базе с помощью внедрения SQL-кода при условии, что в SQL-запросах сайта используются неотфильтрованные данные, вводимые пользователями. Чтобы это предотвратить, необходимо использовать фильтрацию всех данных, внедряемых в SQL.
* XSS (или Cross-Site Scripting). Вредоносный код внедряется на определенную страницу сайта и взаимодействует с веб-сервером злоумышленника при открытии пользователем страницы. Целью данной атаки является получение доступа к cookies и нарушение работы сайта. В качестве защиты необходимо использовать экранирование и кодирование данных на стороне сервера.
* Веб-шелл. Вредоносный скрипт, который используется для взлома сайтов и серверов с целью получения доступа к файловой системе и базам данных. Внедряется с помощью уязвимости в коде сайта или методом подбора паролей.
* DDoS-атаки. Эти действия злоумышленников заключаются в том, чтобы запустить поток трафика на ресурс, с целью нарушить его работоспособность и ограничить доступ к нему. Для совершения DDoS-атак не обязательно собирать гигантское количество человек, злоумышленники умеют внедрять вирусы в любые устройства: от компьютера до умного чайника, и уже эти гаджеты отправляют запросы для атаки. Для защиты от DDoS-атак рекомендуется использовать проверенные хостинги, либо облачные сервисы по типу Cloudflare, а также маскировку IP-адреса, фильтрацию сетевого трафика – например, блокировку по географическому признаку – и многое другое.

Чтобы сохранить безопасность любого веб-сайта, не используйте простые пароли. Злоумышленники используют распространенные способы составления паролей, а также собирают информацию о вас и вашей фотостудии, и используют её в подборе пароля. Подбор может быть сделан с помощью специальных программ, которые автоматически генерируют распространенные слова и сочетания всевозможных символов, либо могут быть использованы данные из соц. сетей или другой раскрытой личной информации в сети.

Чтобы создать надежный пароль, следуйте следующим рекомендациям:

* Длина пароля от 10 символов;
* Не используйте распространенные слова и последовательности букв, типа qwerty;
* Используйте в пароле слова, не связанные друг с другом по смыслу;
* Используйте случайные символы;
* Используйте строчные и прописные буквы, цифры, знаки препинания, а также символы;
* Для разных доступов имейте разные пароли;
* Обязательно активируйте двухфакторную аутентификацию везде, где это возможно. В самых важных доступах старайтесь периодически обновлять пароли.



Рисунок 15. Пример безопасного пароля

Можно использовать генераторы паролей, которые случайным образом создадут для вас определенный набор букв, цифр, знаков препинания и символов. Чем больше настроек у такого генератора, тем лучше.

### Надежный хостинг и дополнительная защита

В первую очередь рассматривайте российские хостинги и регистраторы доменов – так вам не смогут перекрыть доступ по причине вашего географического расположения. При выборе хостинга смотрите на то, что он предлагает в качестве защиты от взлома и атак, и происходят ли постоянные обновления сервера. Определите требования для вашего сайта: объем диска, производительность, выделенный IP-адрес, какие ограничения по трафику, какое есть предустановленное ПО, возможности резервного копирования и пр. Если кто-то в этом не разбирается, то можно посмотреть рейтинг популярных российских хостингов и изучить то, что они предлагают.

Проверьте длительность работы в этой сфере, договор и пользовательское соглашение, есть ли лицензии и своё оборудование, настоящий адрес.

Перейдите с FTP на более безопасный протокол SFTP, либо дайте доступ к FTP только определенным IP. Также можно использовать нестандартные порты для подключения. Разные сайты можно разместить на разных аккаунтах хостинга. Установите права на запись и чтение в папках – 755, для файлов – 644. Таким образом можно определить, что редактировать папки и файлы сможет только владелец.

### Бэкапы

Бэкап (backup) — это резервная копия важных данных, сохраненная в специальный файл. Ею пользуются как подстраховкой: если что-то сломается, с помощью бэкапа можно будет все вернуть. Простейший пример бэкапа — сохранение файла в [облаке](https://blog.skillfactory.ru/glossary/oblachnoe-hranilishe/) или отправка себе в соцсети.

Более сложный пример — резервное копирование образа [операционной системы](https://blog.skillfactory.ru/glossary/operaczionnaya-sistema/), перед тем как она обновится. Например, резервную копию данных перед обновлением предлагают сделать некоторые мобильные ОС. А некоторые делают ее автоматически.

Бэкапы можно делать для сайтов, файлов, систем, программ и так далее.

Обычно бэкап создается автоматически с помощью специальных инструментов. Человек указывает, какие именно важные данные он хочет сохранить в резервную копию, и запускает процесс создания. Нужная информация собирается и архивируется в одном файле.

Если что-то случится, с помощью бэкапа можно будет вернуть потерянные данные, восстановить систему или файл.

Для чего нужны бэкапы

Бэкапы вообще нужны, чтобы избежать необратимой потери данных. Причины могут быть разными:

* файл забыли сохранить;
* в систему неправильно установилось обновление;
* произошла внутренняя ошибка;
* сломался жесткий диск;
* файл заразили вирусы;
* к сервису пропал доступ;
* кто-то случайно удалил данные, и так далее.

Во всех этих случаях информацию можно восстановить из бэкапа — если он актуальный и хранится в надежном месте.

Бэкапы в коммерческой разработке обычно используются для резервного копирования сайтов и сервисов. Перед каким-то важным изменением структуры разработчики создают резервную копию и сохраняют в надежном месте. Если изменение приведет к ошибке, благодаря бэкапу систему можно будет вернуть в норму – в состояние до модификации.

Есть и регулярные бэкапы — их делают на всякий случай, потому что поломки могут быть неожиданными.

Бэкапы нужны для того, чтобы в случае взлома и прочих поломок вы смогли восстановить работоспособность сайта. Сделать бэкап сайта можно как вручную (через FTP-клиент и phpMyAdmin), так и с помощью услуги регулярного резервного копирования у хостинга. Хранить бэкап рекомендуется в другом надежном месте в зашифрованном виде, например в облачных хранилищах, арендованном сервере или у себя на компьютере.

Регулярность бэкапа может быть разной – в зависимости от ваших нужд и предпочтений. Обычно на хостинге они хранятся около 2 недель. Если у вас небольшой сайт, то бэкапы можно делать ежедневно и хранить ежедневные копии за последний месяц. Проверка сайта на вирусы

Для проверки сайты на вирусы сначала рекомендуем проверить, есть ли какая-либо информация об этом в панелях Яндекс.Вебмастер (в разделах «Диагностика», «Безопасность и нарушения») и Google Search Console (в разделе «Проблемы безопасности»). Если информация есть, ждите снижения трафика и падение позиций.

Также у поисковых систем есть безопасные просмотрщики, которые покажут вам статус сайта, т.е. обнаружен ли поисковой системой на сайте небезопасный контент. Для Яндекса инструмент называется Яндекс «Проверка статуса сайта», для Google – «Безопасный просмотр: статус сайта».

Заключение.

Разработанный сайт носит информационный характер. Сайт позволяет всем желающим увидеть информацию о работе фотостудии в любое время и в любом месте, где есть интернет. В дальнейшем будет планироваться модернизация сайта:

* Дополнение новыми функциями;
* Обновление дизайна;
* Расширение прайс-листа;
* Введение обсуждений и так далее.

В результате исследования и разработки, были достигнуты цель и поставленные задачи – данного дипломного проекта:

Цель: Разработка web-сайта для фотостудии «Фото+ г. Королев».

Для достижения поставленной цели были решены следующие задачи:

1. Произвел анализ средств разработки web-сайтов;
2. Определил основные элементы сайта;
3. Выбрал средства реализации для разработки сайта фотостудии;
4. Разработал сайта-визитки.

Сайт – это хороший рекламный инструмент, для продвижения фотостудии. Ведь очень сложно представить любую компанию без него. Эта работа, научила и показала нам, как правильно создавать сайты. Каждый может научиться создавать подобные работы/сайты, стоит только захотеть. Сам дизайн сайта поможет отточить умения и навыки в разработке и оказаться среди лучших. Продолжайте совершенствоваться в данном направлении и не сдавайтесь.

Список используемых источников

* 1. Профессиональная разработка сайтов на Drupal 7; Питер - Москва, 2013. - 688 c.
  2. Байков В. Интернет. Поиск информации и продвижение сайтов; Книга по Требованию - Москва, 2012. - 288 c.
  3. Венедюхин Александр , Воробьев Андрей Создание сайтов (+ CD-ROM); Эксмо - Москва, 2011. - 528 c.
  4. Гарднер Л., Григсби Д. Разработка веб-сайтов для мобильных устройств; Питер - Москва, 2013. - 448 c.
  5. Дакетт Джон HTML и CSS. Разработка и дизайн веб-сайтов (+ CD-ROM); Эксмо - Москва, 2013. - 480 c.
  6. Дронов В. PHP 5/6, MySQL 5/6 и Dreamweaver CS4. Разработка интерактивных Web-сайтов; БХВ-Петербург - Москва, 2009. - 544 c.
  7. Дронов Владимир HTML 5, CSS 3 и Web 2.0. Разработка современных Web-сайтов; БХВ-Петербург - Москва, 2011. - 416 c.
  8. Дронов Владимир PHP, MySQL и Dreamweaver MX 2004. Разработка интерактивных Web-сайтов; БХВ-Петербург - Москва, 2010. - 448 c.
  9. Дронов Владимир РНР 5/6, MySQL 5/6 и Dreamweaver CS4. Разработка интерактивных Web-сайтов; БХВ-Петербург - Москва, 2009. - 544 c.
  10. Дронов, Владимир Macromedia Dreamweaver 4: разработка Web-сайтов; M.: БХВ - Москва, 2014. - 608 c.
  11. Китинг, Джоди Flash MX. Искусство создания web-сайтов; ТИД ДС - Москва, 2012. - 848 c.
  12. Костин С. П. Самоучитель создания Web-сайтов; Триумф - Москва, 2009. - 176 c.
  13. Кристофер Б. Джонс 140 технологий раскрутки сайтов; Рид Групп - Москва, 2011. - 352 c.
  14. Кузнецов М., Симдянов И. PHP. Практика создания Web-сайтов; БХВ-Петербург - Москва, 2012. - 347 c.
  15. Лавдей Ланс , Нихаус Сандра Проектирование прибыльных веб-сайтов; Манн, Иванов и Фербер - Москва, 2011. - 256 c.
  16. Мерсер, Дэвид Drupal 6. Создание надежных и полнофункциональных веб-сайтов, блогов, форумов, порталов и сайтов-сообществ; М.: Вильямс - Москва, 2009. - 272 c.
  17. Митчелл, Скотт 5 проектов Web-сайтов от фотоальбома до магазина; М.: НТ Пресс - Москва, 2013. - 224 c.
  18. Прохоренок Николай Разработка Web-сайтов с помощью Perl и MySQL; БХВ-Петербург - Москва, 2009. - 550 c.
  19. Тероу Шэри Видимость в Интернете. Поисковая оптимизация сайтов; Символ-Плюс - , 2009. - 288 c.
  20. Фрейен Бен HTML5 и CSS3.Разработка сайтов для любых браузеров и устройств; Питер - Москва, 2014. - 304 c.
  21. Фридман В. А., Александров А. В., Сергеев Г. Г., Костин С. П. Строительство Web-сайтов (+ CD-ROM); Триумф - Москва, 2011. - 288 c.
  22. Хуторской А. В., Орешко А. П. Технология создания сайтов. 10-11 классы; Дрофа - Москва, 2011. - 256 c.
  23. Чебыкин Ростислав Разработка и оформление текстового содержания сайтов; БХВ-Петербург - Москва, 2014. - 528 c.
  24. Энж Эрик , Спенсер Стефан , Фишкин Рэнд , Стрикчиола Джесси SEO - искусство раскрутки сайтов; БХВ-Петербург - Москва, 2011. - 592 c.
  25. Энж Эрик , Спенсер Стефан , Фишкин Рэнд , Стрикчиола Джесси SEO. Искусство раскрутки сайтов; БХВ-Петербург - Москва, 2014. - 668 c.