**IT-Колледж “Сириус”**

**\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_**

**ДОКЛАД**

на тему “Веб разработка”

Выполнил:  
Студент группы

К070923/3  
Мокроусов Арсений Владимирович

Принял:

Старший преподаватель  
Тенигин Альберт Андреевич

IT-Колледж “Сириус”  
2023

**Содержание**

* Что такое веб-разработка и немного истории……………..……..……….3
* Фронтенд……………………………………………………………………4
* Бэкенд………………………………………………………….....................5
* Деплой………………………………………………………………………6
* Список литературы ………………………………………………………12

**Веб-разработка**

Я хотел бы поговорить с вами о популярном и востребованном направлении в IT — веб-разработке и разобраться, что это такое, какие существуют ответвления, а также подробно рассмотреть каждое из них.

**Что такое веб-разработка?**

Веб-разработка – это процесс создания веб-приложения или сайта. Существует четыре основные этапа: разработка фронтенда, разработка бэкенда и размещение в интернете. Позже мы подробно разберём каждый из этапов. Но сначала немного из истории.

**Немного истории**

Веб-разработка началась в 1990-х годах, когда только появился интернет. Поначалу сайты выглядели очень плохо. Бедная разметка, файловая система вместо базы данных, статичные страницы – все это относится к времени web 1.0. Чуть позже, в 1999 году началась новая эра Интернета – web 2.0. Отличалась она тем, что пользователи могли управлять контентом – создавать, редактировать и удалять. В такое время разработкой занимался веб-мастер. Он немного владел HTML, делал простейший дизайн и чуть-чуть работал с базами данных. Но со временем появились другие специалисты, и профессия веб-мастера полностью исчезла.

Теперь давайте разберём современный процесс разработки сайта. Как я говорил ранее, он делится на 4 этапа.

**Фронтенд**

Фронтенд разработка – это разработка тех частей сайта, которые пользователь будет видеть и взаимодействовать с ними. Процесс такой разработки состоит из двух этапов – написания дизайна и вёрстки страницы.

Дизайн можно сделать тремя способами:

* Самостоятельно
* Заказать у профессионала
* Найти готовый бесплатный вариант

Если вы уверены в том, что сможете сами создать подходящий дизайн своего веб-приложения, то, безусловно, вам подойдет первый способ.

Но бывают те, кто не владеют такими навыками, либо просто не хотят. Таким разработчикам подойдет второй способ.

Ну а если вам вообще не важно то, как будет выглядеть ваш сайт, вам отлично подойдет третий способ. Его не рекомендуется использовать при создании серьезного проекта с определёнными целями.

Как только дизайн готов, необходимо воплотить его в жизнь, или, другими словами, наверстать. Вёрстка – это процесс преобразования нарисованного дизайна в полноценную веб-страницу при помощи трёх замечательных инструментов – HTML, CSS и JavaScript.

* HTML (Hyper Text Markup Language (язык гипертекстовой разметки)) – это технология, позволяющая задать структуру расположения элементов веб-страницы. Не является языком программирования.
* CSS (Cascading Style Sheets (каскадные таблицы стилей)) – это технология, позволяющая изменить отображение HTML документов и свойств конкретных элементов на веб-странице.
* JavaScript - язык программирования, который позволяет придать веб-страницам интерактивность при помощи взаимодействия с DOM (Document Object Model (объектная модель документа)). На основе JS написаны некоторые фреймворки, такие как Vue.js, Angular, и др. Сейчас мы разберем первые два. Однако также мы рассмотрим React, хоть он и является библиотекой JS, а не фреймворком, все же многие фронтендеры используют его для разработки.
  + Vue.js – выделяется своей простотой в изучении, масштабируемостью, а также богатой библиотекой. Из минусов можно выделить небольшое комьюнити и чрезмерную гибкость. Последнее означает, что на Vue можно писать почти, как угодно, а это значит, что в большом проекте может возникнуть проблема целостности кода. Другими словами, один фронтендер реализовал компонент так, а другой фронтендер реализовал другой компонент иначе, и при интеграции этих компонентов в общий проект при их взаимодействии могут возникать проблемы.
  + React – обладает большим комьюнити, суперской оптимизацией, возможностью создавать многоразовые компоненты, совместимостью со старыми версиями, а также виртуальной DOM. В минусы входят бедная библиотека официальных модулей, отсутствие конвенций (стандартов разработки), а также сложность в изучении новичку.
  + Angular – отдельно поддерживает TypeScript, обладает подробной документацией, структурой и архитектурой, обеспечивающей масштабируемость, а также большой библиотекой. Недостатками являются слишком высокая мощность для небольших проектов и производительность.

**Бэкенд**

Бэкенд разработка – это разработка на стороне сервера. Заключается в разработке тех частей сайта, которые пользователь не видит. В такой разработке используются технологии, такие как:

* PHP
* Ruby
* Python

**PHP** – язык программирования, активно используемый в бэкенд разработке. Подходит для изучения азов веб программирования. Отличается своим низким порогом входа. Это означает доступность и простоту использования. Зачастую разработка на PHP проходит при помощи гибкого open-source фреймворка Laravel. Он позволяет разрабатывать сложные веб-приложения гораздо быстрее и безопаснее, чем другие фреймворки благодаря своей богатой библиотеке готовых пакетов (ORM, механизмы аутентификации, тестирование, и т. д.), высокой загрузке страниц, а также огромным сообществом разработчиков по всему миру. Компании, использующие PHP: Facebook, Viber, Lyft, Mint.

**Ruby** – язык программирования общего назначения, зачастую используемый в веб разработке. Его особенностями являются простота, доступность, интерпретируемость, а также сильный уклон в ООП. В нем очень много функций для работы с объектами. Разработка на нем проходит при помощи фреймворка Ruby on Rails. В преимущества RoR входят: богатая библиотека инструментов, масштабируемость, огромное сообщество, строгое соблюдение стандартов. Масштабируемость позволяет реализовывать проекты любого масштаба. Примеры сайтов, использующих Ruby: Codecademy, Airbnb, TaskRabbit.

**Python** – интерпретируемый язык программирования общего назначения. Используется в работе с большими данными, машинном обучении и веб разработке. Имеет простой синтаксис, огромнейшую библиотеку, а также совместимость с другими языками. Вот примеры сайтов на Python: Uber, Pinterest, Spotify, Mozilla, Netlify. Самый популярный и востребованный фреймворк для веб разработки – Django. Он содержит очень большое количество инструментов для разработки, такие как ORM, аутентификация, формы, панель администратора и т. д. Его принцип устройства позволяет очень гибко настраивать проект. Одной из таких особенностей можно выделить *приложения*. Они позволяют разделить проект на несколько частей, что существенно ускоряет разработку и повышает читаемость кода. Еще один не менее популярный фреймворк – Flask. Он масштабируемый, минималистичный, простой в использовании. Однако у него ограниченный функционал, то есть большинство функций приходится писать самому. Говорю по своему личному опыту. Я разрабатывал социальную сеть на Flask-е, и мне много чего приходилось разрабатывать самостоятельно.

Ниже представлена статистика использования backend-фреймворков на январь 2023 года (рис. 1). Как видим, на первом месте находится Lavarel, язык PHP. Второе и четвертое места заняли Python фреймворки – Flask и Django. А на третьем Spring. Это фреймворк, написанный на языке Java. Из плюсов можно выделить огромную базу инструментов, большое сообщество, доступность, масштабируемость, ускорение работы. К минусам относятся долгая настройка, сложное начало, большой вес кода. И только на 7 месте рассмотренный ранее Ruby on Rails.

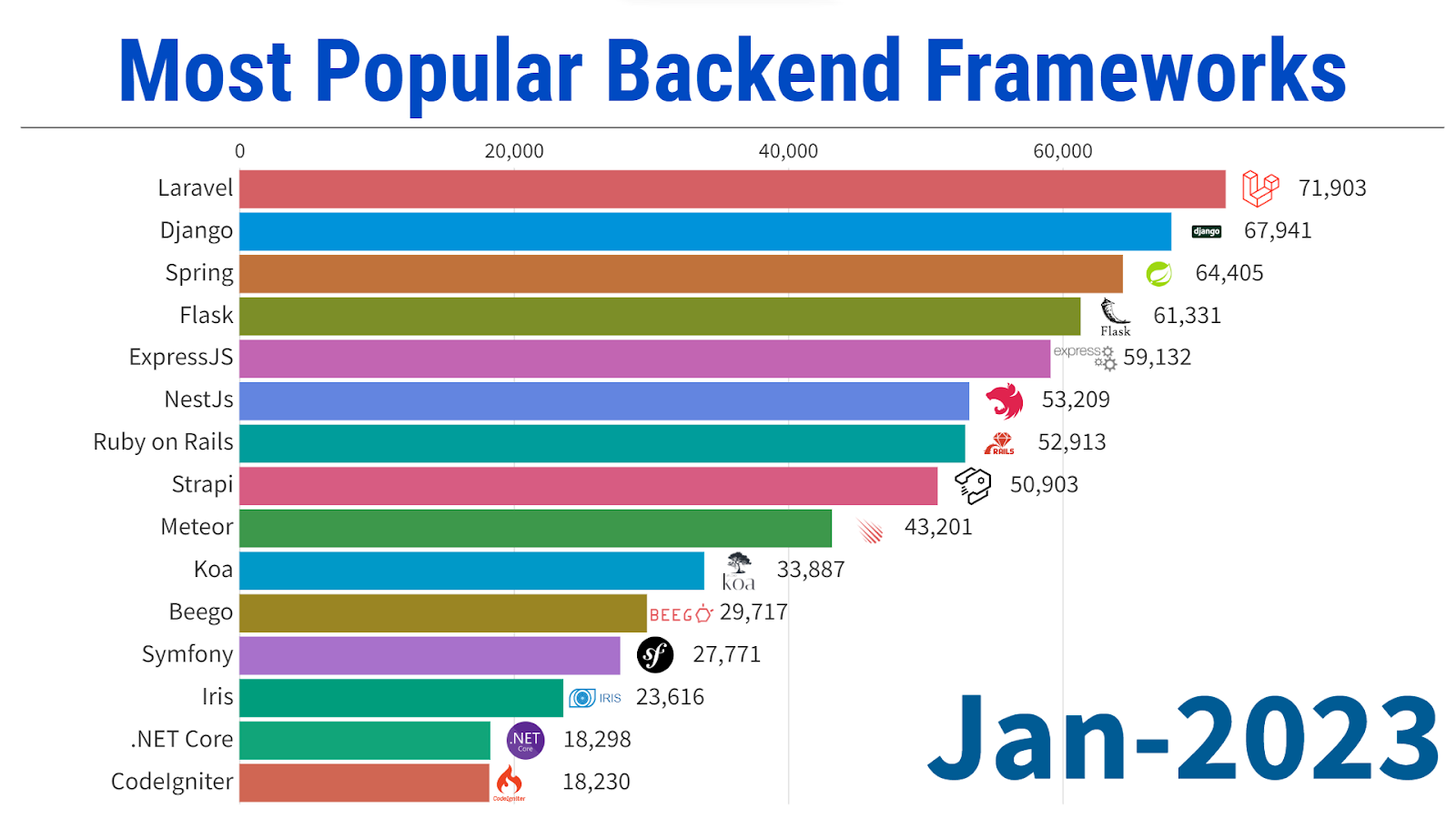


Рис.

**Деплой**

В процессе разработки проекта все наработки и плоды вашего труда рекомендуется загружать в репозиторий GitHub. Данный сервис используют для хостинга проектов и совместной разработки в больших IT-компаниях. Кроме того, вы можете посмотреть проекты других людей и узнать что-то новое из них, либо преподнести кому-то урок своим проектом.

После того, как работа над проектом закончена и вы готовы представить его миру, необходимо сделать деплой. Деплой – это развертывание проекта на веб-сервере или хостинге. Его сложность зависит от используемых инструментов. Для реализации такой процедуры программист совершает следующие действия:

* Код проекта загружается на сервер/хостинг
* Устанавливаются все зависимости
* Инициализируется база данных
* Производится сборка и запуск программы

Деплой производят на специальную виртуальную машину, которую предоставляет хостинг-провайдер. Хостинг – это услуга, сервис, который выделяет вам «кусочек» сервера, а также необходимый набор инструментов для управления таким кусочком. Сервер находится онлайн круглосуточно, что позволяет пользоваться вашим проектом в любое время суток.

Рассмотрим варианты хостингов:

1. Виртуальный хостинг (Shared) – бюджетный вариант для вашего проекта. Вам выделяется какая-то часть ресурсов большого сервера, держащего множество проектов других пользователей. Включает в себя сервер с заранее настроенным оборудованием под определенный стек, что существенно упрощает процесс деплоя. Из плюсов можно выделить стоимость и уже настроенное оборудование. Из минусов выделим ограниченность ресурсов, а отсюда невозможность поддержки большого потока посетителей.
2. VPS (Virtual Private Server) ― виртуальный выделенный сервер. Также его называют VDS (Virtual Dedicated Server). Отличия лишь в названиях и технологиях виртуализации: VDS использует аппаратную виртуализацию, а VPS – на уровне ОС. Такой вид хостинга очень похож на виртуальный хостинг. Однако здесь у пользователя есть возможность в установке своего ПО, а ограничение лишь в предоставляемых вам ресурсах. Плюсами являются масштабируемость и то, что выделенные ресурсы полностью в вашем распоряжении. Минусы – стоимость.
3. Выделенный хостинг (dedicated server) – это полноценный компьютер для вас. Сюда можно поставить любую ОС и ПО. Плюсами являются максимальные возможности настройки сервера, а минусами – цена.
4. Облачный хостинг – это хостинг, использующий не один, а несколько своих серверов для предоставления услуг, что равномерно распределяет нагрузку на оборудование и, как следствие, увеличивает отказоустойчивость. Из плюсов – платите только за те ресурсы, которые использовали, а также надежность. Из минусов – нестабильность цены.

Каждый вид хостинга специализируется на своем стеке задач. Чем масштабнее проект, тем больше ресурсов и возможностей управления нужно.

**Мобильная разработка**

С каждым днём в интернете появляется все больше и больше новых пользователей. Кто-то серфит через телефон, а кто-то через компьютер. Но ведь экраны у всех устройств отличаются по размерам. Соответственно, и внешний вид сайта может отличаться на разных устройствах.

Ниже (рис. 2) мы видим статистику использования интернета по всему миру на январь 2023 год. Каждый год ее составляют и публикуют несколько компаний - сервис Datareportal, креативное агентство We Are Social и платформа медиа-аналитики Meltwater. Первый столбик содержит общую численность населения и процент урбанизации. Во втором столбике мы видим общее число пользователей мобильного телефона в интернете, а также процент их популярности среди всех пользователей интернета. В третьем столбце указано количество пользователей интернета и их процент от всего населения. И последний столбик говорит нам о пользователях социальных сетей – их число и процент от всех пользователей интернета.

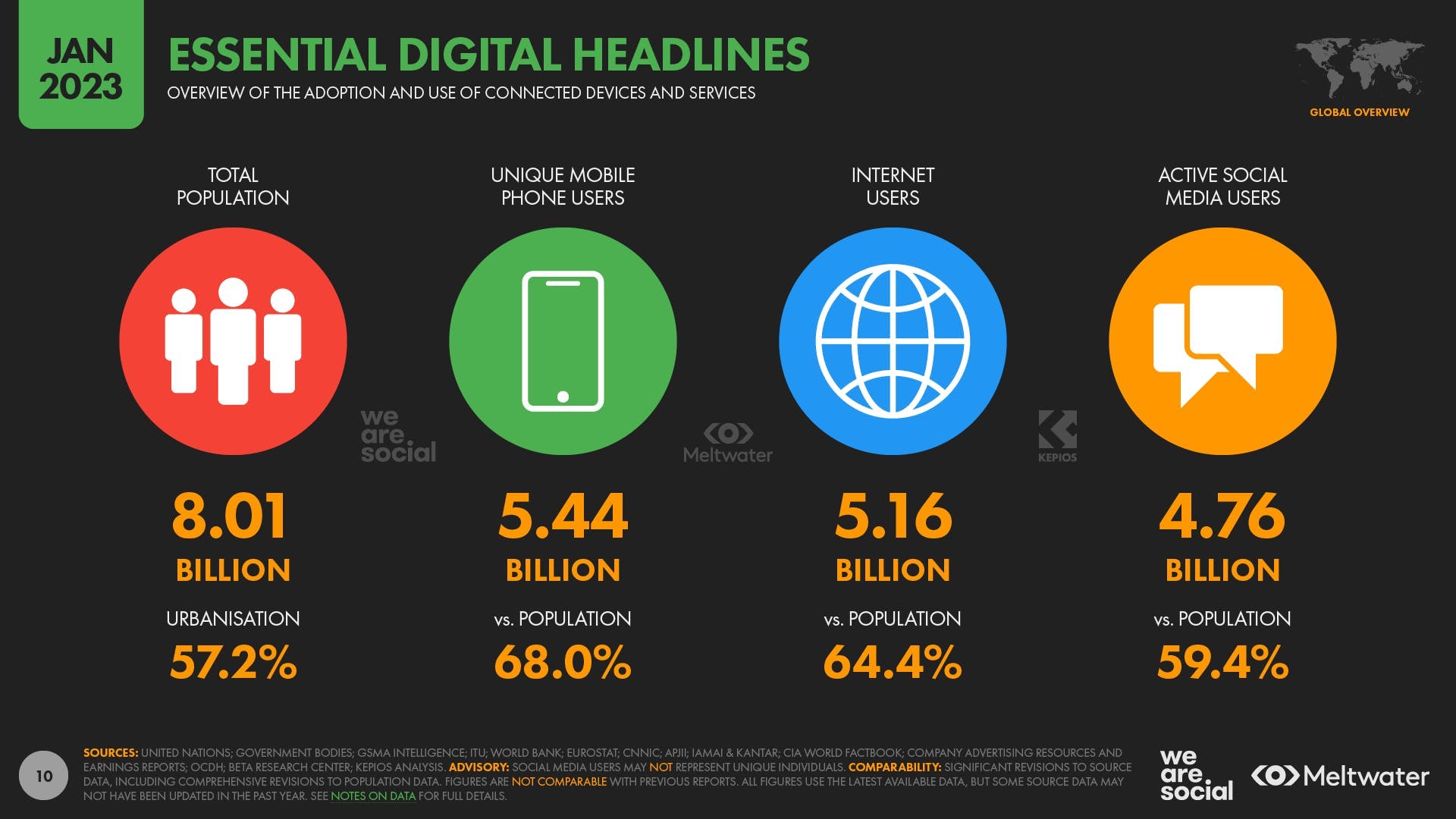


Рис. 2

Из этой статистики мы можем понять, что большинство юзеров используют мобильный телефон для доступа в интернет. А это значит, что нужно адаптировать сайт не только под большие экраны компьютеров и планшетов, но и под такие маленькие экраны.

При мобильной разработке веб-приложений необходимо учитывать несколько особенностей:

* Разнообразие экранов. На рынке существует огромное количество устройств с разными экранами. В связи с этим необходимо разработать веб-приложение, учитывающее данный фактор.
* Ограничение ресурсов. Многие пользователи используют телефоны с устаревшими комплектующими. Разработчику нужно максимально оптимизировать его проект, чтобы удовлетворить потребности наибольшего количества посетителей.
* Мобильные браузеры. Веб-приложение должно уметь взаимодействовать со всевозможными браузерами и обеспечивать наилучшую работу с ними.

На протяжении многих лет сайты имели либо фиксированную верстку (при любом раскладе ширина оставалась одной), либо резиновую (при изменении ширины окна элементы растягивались или сжимались). Но такие подходы устарели, ведь такие сайты некомфортны в использовании. И придумали 2 вида верстки – адаптивную и отзывчивую.

* **Адаптивная верстка.** Верстка, при которой сайт имеет несколько вариантов отображения под разные экраны. Например, при посещении сайта с планшета он отображается так, а если зайти с компьютера, уже иначе. Такой подход позволяет комфортно использовать ваше веб-приложение. Ведь, очевидно, с компьютера будет удобнее читать текст в 20 пунктов, а не 14.
* **Отзывчивая верстка.** Здесь будет немного иначе. При изменении ширины окна элементы будут изменяться при помощи анимации. Это также позволит посетителям испытывать удовольствие, используя ваш сайт.

Таким образом, при верстке сайта необходимо также владеть хотя бы одним из этих подходов. Так вы будете удовлетворять потребности каждого пользователя с любым экраном.

**Выводы**

* Веб-разработка – это процесс создания сайта или веб-приложения. Состоит из 3 этапов – разработка фронтенда, разработка бэкенда и деплой.
* Существует 3 поколения сайтов – web 1.0, web 2.0 и web 3.0.
* Фронтенд разработка – это создание внешнего вида сайта. Разрабатывается при помощи HTML, CSS и JS.
* Бэкенд разработка – это создание функциональной части сайта. Разрабатывается при помощи Python, PHP, Ruby, и не только.
* Деплой – это развертывание веб-приложения на хостинг. Существует 4 вида хостингов – shared, VPS (VDS), dedicated и облачный.
* Большинство пользователей используют мобильные устройства, а это значит, что необходимо адаптировать фронтенд не только под большие экраны, но и под маленькие. Для этого существуют два подхода верстки – адаптивный и отзывчивый.

**Список литературы**

* Михаил Русаков – Создание сайта от начала до конца
* Джон Дакетт – JavaScript и JQuery. Интерактивная веб-разработка
* Алексей Гладкий – Как самому создать сайт быстро и профессионально
* What is Web Development? The Ultimate Guide – Jenna Inouye (Hackr)
* Что такое веб-разработка: основы, технологии и специализации – Вадим Пашаев (PXSTUDIO)
* Видит сайт насквозь: чем занимается веб‑разработчик и как им стать – Алексей Гмитрон, Александра Патрушева, Полина Овчинникова (Яндекс Практикум)
* Зачем нужен PHP – Игорь Антонов (Журнал «Доктайп»)
* 15 причин выбрать Laravel для разработки сайта – Красный Хамелеон
* Плюсы и минусы Ruby on Rails в 2023 – Блог UniwexSoft
* Ruby – Блог SkillFactory
* Python – Блог SkillFactory
* Spring - Блог SkillFactory
* В Java всегда весна: почему фреймворк Spring считается самым популярным у Java‑разработчиков – Пётр Кушнир, Лена Шпрингер (Яндекс Практикум)
* Pros and Cons of Django as web framework for Python developers - Michał Macura (Перевод от germn – Плюсы и минусы Django (Хабр))
* Most Popular Backend Frameworks – 2012/2023 – Statistics and Data
* Хочу создать сайт. Как выбрать подходящий хостинг – 2domains
* Что такое деплой в программировании - Digital-агентство Prostudio (Яндекс Практикум)
* Чем отличается VDS от VPS-сервера - Команда Timeweb Cloud
* DIGITAL 2023: GLOBAL OVERVIEW REPORT – DataReportal
* Что такое Vue.js – AppMaster
* Vue.JS: особенности, преимущества и недостатки – Крым Диджитал
* React или Angular или Vue.js — что выбрать? – yalatan (Хабр)
* React: плюсы и минусы популярной библиотеки JavaScript - Nuances of programming (Дзен)
* Плюсы и минусы React JS в 2023 – UniwexSoft
* Angular плюсы и минусы в 2023 - UniwexSoft