**РАЗРАБОТКА ВЕБ-СТРАНИЦЫ «ЛУЧШИЕ НЕЙРОСЕТИ НАШЕГО ВРЕМЕНИ» С ИСПОЛЬЗОВАНИЕМ ТЕХНОЛОГИЙ HTML И CSS**

Пояснительная записка к курсовому проекту

НАТК ПР-22.103к

Разработал:

Шатровой Е.С.

* ВВЕДЕНИЕ

Данный курсовой проект является актуальным, так как по анализу интернет-ресурсов, было выявлено, что красивых и удобных сайтов для просмотра информации о нейросетях крайне мало, поэтому есть необходимость в создании платформы для вхождения в мир нейросетй.

Наименование программного продукта «Лучшие нейросети нашего времени». Продукт представляет собой одну веб-страницу, предназначенную для бесплатного сбора базовой информации о нейронных сетях.

Разрабатываемая веб-страница позволит пользователям изучить информацию о нейронных сетях. Предполагаемая потребность обусловлена тем, что при поиске необходимой информации, не было обнаружено одновременно красивой и удобной в изучении веб-страницы.

Целью курсовой работы является разработка сайта с использованием технологий HTML и CSS.

Для достижения цели необходимо выполнить следующие задачи:

* проанализировать образовательные платформы по совеременным нейронным сетям;
* спроектировать дизайн сайта с учётом информации о нейросетях и их перспективах в будущем;
* изучить такие языки программирования и разметки, как HTML, CSS, JavaScript;
* изучить принцип создания веб-страниц.

Объектом исследования курсовой работы являются нейросети и технологии фронтенд разработки.

* ВЫБОР ТЕХНОЛОГИИ, ЯЗЫКА И СРЕДЫ ПРОГРАММИРОВАНИЯ

Данная веб-страница использует такую среду программирования, как:

Sublime Text 3 – это редактор исходного кода, кроссплатформенный текстовый редактор для написания программного кода на различных языках программирования (Groovy, Erlang, C+, Java и т.д.), а также верстки веб-документов.

Также будет использоваться такая библиотека, как:

JQuery – набор функций JavaScript, фокусирующийся на взаимодействии JavaScript и HTML. Библиотека jQuery помогает легко получать доступ к любому элементу DOM, обращаться к атрибутам и содержимому элементов DOM, манипулировать ими. Из языков программирования будут необходимы:

Html – это стандартный язык разметки гипертекста в Интернете. Его основное предназначение – создавать интернет-страницы и обеспечивать нормальное расположение в документе списков, заголовков, таблиц, картинок и прочих материалов.

Css (Cascading Style Sheets, каскадные таблицы стилей) – это язык описания внешнего вида HTML-документа. Это одна из базовых технологий в современном интернете. Практически ни один сайт не обходится без CSS, поэтому HTML и CSS действуют в единой связке.

JavaScript – это мультипарадигменный язык программирования. Поддерживает объектно-ориентированный, императивный и функциональный стили. JavaScript обычно используется как встраиваемый язык для программного доступа к объектам приложений. Наиболее широкое применение находит в браузерах как язык сценариев для придания интерактивности веб-страницам.

* Проектирование дизайна веб-страницы

Для разработки мной был выбран сайт-одностраничник.

При загрузке сайта в центре необходимо показать только его название и пару строк с его описанием, что бы пользователь точно и ясно понял, находтся ли на этой странице нужная ему информация.

После прочтения краткого содержания сайта пользователю потребуется узнать какие темы на нём обозреваются, понадобится небольшая навигационная панель для быстрого перемещения между темами и стрелка для мгновенного перемещения к панели.

Дальше идёт блок с информацией о назначении нейронных сетей, их создании и применении в нашей повседневной жизни.

В дополнение к предыдущему блоку приводится список лучших нейросетей художников предназначенных для работы с графическими изображениями. На их примере проще всего понять принцип их работы и генерации результатов. Если пользователя заинтересует одна из нейросетей, то он может без труда перейти на официальный сайт нейронной сети и воспользоваться ею самостоятельно.

Что бы не создавалось впечатления, будто нейросети способны только на генерацию изображений было добавлено несколько простых и понятных блоков с видео и примерами других нейросетей, в тех же блоках тема медленно сводится к вытеснению человеческого труда. Глядя на 7 блоков, кажется что они занимают слишком много места, было принято решения поместить часть из них в спойлер, который при необходимости можно открыть или закрыть что бы “не мозолил глаза”.

Для любителей музыки (и просто мне хотелось встроить аудио плеер) был добавлен блок с музыкой полностью сгенерированной нейросетями, она не совершенна и придётся по вкусу не всем, но в будущем их композиции станут несколько идеальней.

Для общего понимания каждой статье, работе и сайту нужен итог, который и был написан. В самом конце было добавлено видео с нейронной сетью Neuro-sama, это искуственный интеллект который на видео играет в такие игры как osu!, Minecraft или Slay the Spire . Нейронная сеть имеет собственное мнение, характер. На запущенных стримах она разговаривает и отвечает на вопросы зрителей, на добавленном отрывке видео наглядно показано как она прилюдно унизила одного из зрителей за негативный комментарий её игры.

* HTML разметка сайта

С HTML разметки начинается создание любой страницы, эта не исключение.



Представили структуру сайта в голове и за пару секунд выставили несколько основных блоков, сразу же выставляя им базовые классы (которые пропишем позже) и id для будущей панели навигации.



В “шапку” сайта закидываем фоновое изображение (все стили блоков будут прописаны ниже), для видео указываем атрибуты отвечающие за автовоспроизведение видео, отключение контроллеров и автоматический повтор, чуть ниже указываем название самого видео. В head\_mid в заголовке h1 указан текст заголовка, тегу p (от лени) сразу добавляем стили отвечающие за тёмно синюю разделяющую черту, а так же и сам текст.



Навигационная панель с классом anchor (т.к. в ней находятся якорные ссылки) находятся как ни странно, сами якорные ссылки, по нажатию по которой плавно переносимся к интересующему месту сайта. К некоторым блокам приписаны id=”anchor1”, id=”anchor2” и т.к., а каждая ссылка переносить пользователя к указанному в ней id.



Разметка блока с классом mid слишком огромна, так что рассмотрим только одну горизонтальную секцию состоящую из блока с картинками и текстом. Эти два блока визуально разделены двумя комментариями <!--элемент--!>. Если в первом указан заголовок текст и ссылка на проект, то со вторым всё несколько интереснее. Показать в блоке одну картинку – скучно, сделать слайдер – долго и сложно. А можно сделать слайдер картинок, быстро и просто, нужно просто импортировать в проект библиотеку сделанную другим человеком, а далее вставить в слайдер свои картинки, просто и на удивление работает.



Далее идёт блок с примерами других нейросетей, в HTML разметке это выглядит скучно, по сути коробка с видео, текстом и ссылкой. И так 6 раз. С 7мым поинтереснее. В 7мом блоке есть кое что интересное, если точнее, то импортированное видео с YouTube. Код для платформы предоставляет сама платформа.



Ничего сложного чтобы скопировать код, который выдал тебе сам YouTube.

Последующие 2 блока текста вообще не отличаются новизной. Разве что в первом, да и то код для плеера опять предоставила сама платформа. 

Теперь с HTML разметкой закончили.

* Стили для элементов сайта

Выше озвученный план нужно выполнять в точном порядке, что бы не нарушить логическую цепочку.

Было просмотренно множество вариантов дизайна сайта и его цветовой палитры, дизайн сайта был придуман ещё давно и гениальностью он не отличается, цветовая палитра состоит из нескольких оттенков серого и тёмно синего.

“Шапку” сайта нужно сделать яркой и запоминающейся, для этого на фон должно быть помещено видео, хотя компьютер воспринимает видео не как фон, а как видео отодвинутое на задний план.



Видео был выдан класс video1, с помощью первой строчки убраны отступы, с помощью 2 и 3ей строки закрепили видео в неподвижном положении и “отдали приказ” не реагировать на стили других элементов сайта. Следующие 2 строки растянули видео на всю ширину экрана, а z-index отодвинул видео на задний план.



Краткая информация о сайте (его название и краткое описание) должны помещены по центру экрана и чётко читаться. head – вся шапка, head-mid элемент по центру. display: block; если что то не работает как надо, добавляем эту строку и оно работает (я добавляю только по этой причине, мне помогает). height auto не задаёт конкретную высоту, она даёт то количество пространства которое требует сам блок. padding задаёт внутренний отступ в самом блоке, а margin отступ от других элементов сайта. С помощью вышеприведённого кода, были убраны стандартные отступы, добавлен отступ от “потолка” браузера и отодвинута нижняя панель навигации.



В дальнейшем мне точно понадобятся различные стили для основного текста, заголовков и видео, на тот момент разработки они мне ещё не были нужны, но каждый из них пригодится в будущем и “выстроены” заранее, что бы на них не отвлекаться потом. float указывает к какому краю прижать элемент, в нашем случае текст, font-size указывает размер шрифта в % (правда, мне не ясно в % от чего). У классов .videoN видно несколько подозрительных элементов, такие как :hover и transition, первый отвечает за добавление указанному классу некоторых стилей после наведения на него курсора, второй отвечает за плавную анимацию добавления этих стилей.



Создание панели навигации, через стили она была “растянутянута” на 100% ширина экрана и задана высота в 70 пикселей, атрибут auto добавлен что бы блок мог сам подстраивать свои размеры под его наполнение. Внутри панели навигации находятся ссылки, которые переносят пользователя на определённые участки страницы, в стилях ссылки указан отступ и указана полоса тёмно синего цвет для сигнала “*это ссылка, а не текст*”. Чем различаются два класс этих ссылок вы можете разобраться и без меня.



Класс DopText используется только для отображения ничем не выделяющегося блока текста с указанными отступами, шириной, высотой, текстом фона и тенью от него.

Был создан класс mid с не самыми интересными стилями, соответственно и вставлять их смысла нет.

Для наиболее простого, красивого и интересного представления списка элементов были придуманы таймплайны. Они сложны в создании, так что я помечал код “комментариями”, подписи к коду которые никак не влияют на сам код. Блоки были распределены на горизонтальные секции из двух блоков, с картинкой и текстом. Блоки делятся на чётные(+1) и нечётные(0) блоки, стили для каждого из них несколько отличаются. По середине с краю блоков, за их пределами находится стрелка указывающая на линию прописанную в всё тех же стилях. Таймплайн смотрелся несколько сухо и скучно, был вырезан фон у всего класса mid тем самым “обножая” фоновое видео, установленное и закреплённое ещё в самом начале, дабы видео не сильно отвлекало от чтения оно было затемнено в полтора раза.



Передомной стояла цель сделать несколько блоков с новостями, ничего лучше и проще “видео сверху, текст снизу” придумать не смог, хотя и не пытался. Все 7 блоков за исключением последнего полностью одинаковы по структуре и различаются только содержанием. Каждый имеет ширину 25% от ширины экрана, расположен по левою сторону экрана (наличие этой строки у каждого выстраивает их в линию), отступ слева на 6%, одинаковый фон и тень отблока.

Что касается изменённого последнего 7го блока, он должен в себя вместить видео с ютуба и больше информации, что значит что он занимает больше места по ширине, последние 2 строки position и overflow позволяют масштабировать видео с ютуба в процентах (чего стандартные настройки не позволяют делать), а текст был перемещён из под видео в сторону от него.

Над спойлером для “сокрытия” последних 4х блоков не очень хотелось заморачиваться и думать, за меня давно всё сделано, а код подобных спойлеров выложен в сеть. Стили были скопированы, изменена только анимация раскрытия, цветовая палитра и наполнение.



Сайт и стили к нему полностью сделаны, но есть одна причина почему работа не окончена, хочется анимацию появления блоков добавить. Всегда говорят не изобретать веловипед заново, да и не очень хочется, все функции давно сделанны до меня и перенесены в библиотеки. Одну из таких я и подключу, а после добавлю 4 заветных слова в класс к блокам.



animate\_\_animated отвечает за добавление анимации блоку, а animate\_\_fadeInUp за вид анимации. Всё бы было бы хорошо, если бы не одно НО. Как только мы включаем сайт анимации проигрываются все и сразу, вне зависимости от того видим мы это или нет. Настало время подключить ещё одну библиотеку WOW(и добавить её название в класс к блоку), она работает довольно просто, включает работу скриптов только тогда когда это увидит пользователь.

Вот теперь работа над сайтом полностью завершена, ну или мне лень сделать больше.

* Небольшой итог

Ну, что я могу сказать. Сайт сделан за 2 месяца, если бы не ленился, то можно было бы за +- 2 недели. Научился крайне многому, пользование сайтом Хабр, code.io, Stack Overflow, htmlbook.ru и множество, множество остальных.

+опыт создания сайтов, работа с библеотеками и интеграция кода других пользователей и платформ. Мне понравилось :D

* Ссылки на источники
* <https://codepen.io/>
* <https://ru.stackoverflow.com/>
* <https://www.internet-technologies.ru/>
* <https://qna.habr.com/>
* [https://htmlbook HYPERLINK "https://htmlbook.ru/" HYPERLINK "https://htmlbook.ru/" HYPERLINK "https://htmlbook.ru/" HYPERLINK "https://htmlbook.ru/" HYPERLINK "https://htmlbook.ru/" HYPERLINK "https://htmlbook.ru/" HYPERLINK "https://htmlbook.ru/". HYPERLINK "https://htmlbook.ru/" HYPERLINK "https://htmlbook.ru/" HYPERLINK "https://htmlbook.ru/" HYPERLINK "https://htmlbook.ru/" HYPERLINK "https://htmlbook.ru/" HYPERLINK "https://htmlbook.ru/" HYPERLINK "https://htmlbook.ru/"ru/](https://htmlbook.ru/)
* <https://www.iphones.ru/iNotes/591360>
* <https://vk.com/video/playlist/-168874636_3>
* <https://webkab.ru/>
* <https://thecode.media/neuro-2022/>
* <https://pikabu.ru/story/podborka_topovyikh_besplatnyikh_neyrosetey_kotoryie_mozhno_poprobovat_uzhe_seychas_9587266>
* <https://wowjs.uk/docs.html>
* <https://animate.style/>
* <https://github.com/animate-css/animate.css>
* <https://github.com/animate-css/animate.css>
* <https://animate.style/>