

09.02.07 ИНФОРМАЦИОННЫЕ СИСТЕМЫ И ПРОГРАММИРОВАНИЕ

Квалификация: разработчик веб и мультимедийных приложений

Я выбрал свой путь — веб-разработка. Что ждет меня впереди?

Веб-разработчик занимается созданием сайтов и веб-сервисов.

Какие задачи решает веб-разработчик

Когда говорят “веб-разработчик”, как правило, имеют в виду фронтендера, который разбирается в бэкенде.

Чем отличается фронтенд и бекенд разработчики?

Фронтенд

Чтобы представить контент сайта наилучшим образом и обеспечить идеальное взаимодействие пользователя с сайтом, фронтенд-разработчики используют такие инструменты как:

- HTML (HyperText Markup Language) — язык разметки документов для создания структуры страницы: заголовки, абзацы, списки и так далее.

- CSS (Cascading Style Sheets) — язык для описания и стилизации внешнего вида документа. Благодаря CSS-коду браузер понимает, как именно отображать элементы. CSS задаёт цвета и параметры шрифтов, определяет, как будут располагаться разные блоки сайта, и так далее. Ещё он позволяет выводить один и тот же документ в разных стилях, например, для печати (обычной или шрифтом Брайля), вывода передачи на экран или чтения голосом.

- JavaScript — это язык, который создавался, чтобы оживить веб-страницы. Его задача — реагировать на действия пользователя, обрабатывать клики мышкой, перемещения курсора, нажатия клавиш. Ещё он посылает запросы на сервер и загружает данные без перезагрузки страницы, позволяет вводить сообщения и многое другое.

- Ряд других элементов и фреймворков (SAAS, Bootstrap, Vue.js, React.js и т.д.) используют вместе с основными инструментами, указанными выше, для значительного облегчения работы.

Бэкенд-разработка: «сердце» сайта

Бэкенд - «невидимый двигатель» сайта. Бэкенд-разработчики пишут код, используя такие популярные языки программирования: PHP, Ruby on Rails, Python, NET и другие. Когда нужно выполнить какую-то операцию, бэкенд-код взаимодействует с системами управления базами данных, например:

- PostgreSQL;
- SQLite;
- MongoDB.
- MySQL,
- SQL.

После этого необходимая информация возвращается пользователю в форме фронтенд-кода.

Заниматься онлайн-шопингом? Подписываемся на рассылку? Редактирование контента на странице? Какую бы операцию вы не выполняли, в конце концов за нее отвечает бэкенд-код. Информация обновляется, изменяется или удаляется в базе данных.

Веб-разработчик создает новые сервисы и сайты или поддерживает и дополняет существующие. Он изучает задачу заказчика и помогает создать оптимальное решение, готовит код для фронтенда и бэкенда, тестирует его, исправляет ошибки и оценивает результаты. Вот, например, какие задачи он может решать:

Веб-разработчик тесно взаимодействует с дизайнером, менеджером и другими разработчиками.

Этапы работы и инструменты.

Представим себе, что нужно запустить онлайн-магазин. Сначала разработчик получает задачу и оценивает сроки ее выполнения. Здесь он может обсуждать решение с техническим директором или с тимлидом. Если сроки всех устраивают, их фиксируют в трекере задач. Например, в Trello, Яндекс.Трекер или YouTrack.

Новый онлайн-сервис — это большая задача. Чтобы выдержать сроки, ее нужно разбить на более мелкие — декомпозировать. Если стоит задача разработать отдельную функциональность для существующего сервиса, то всё начинается с работы с дизайнером. Требуется изучить макет и прояснить все детали решения. Для этого разработчик пользуется графическими редакторами, обычно это Sketch или Figma.

Следующий этап — написание кода. Писать код можно хоть в блокноте, но чаще это делают в специальном текстовом редакторе или в IDE. Самые популярные — VS Code и WebStorm.

Когда код написан, нужно его протестировать. Например, проверить, не сломается ли функциональность, если пользователь сделал что-то, чего мы от него не ждали. Кроме ручного тестирования иногда веб-разработчик пишет автоматизированные тесты. Они не требуют участия человека, поэтому их можно запускать часто. Это позволяет отслеживать ошибки, которые могут возникать при внесении изменений в код. Чтобы не потерять замечания коллег и зафиксировать все этапы работы, команды пользуются инструментами контроля версий и совместной работы над кодом — Git, Github.

С чего начнется путь разработчика веб и мультимедийных приложений в техникуме?

Первый курс (1-2 семестры) — это общеобразовательные предметы (10-11 класс). Например, математика, информатика, экология и т.д.

Второй курс — это начало пути развития веб-разработчика. Начинается путь с понятия алгоритмизации и основ программирования.

Зачем веб-разработчику знать алгоритмы и применять их на практике?

— умение эффективно решать поставленные задачи и не изобретать велосипедов,

— умение не запуская код доказывать работоспособность кода,

— умение эффективно и с максимальными возможностями использовать выбранный язык,

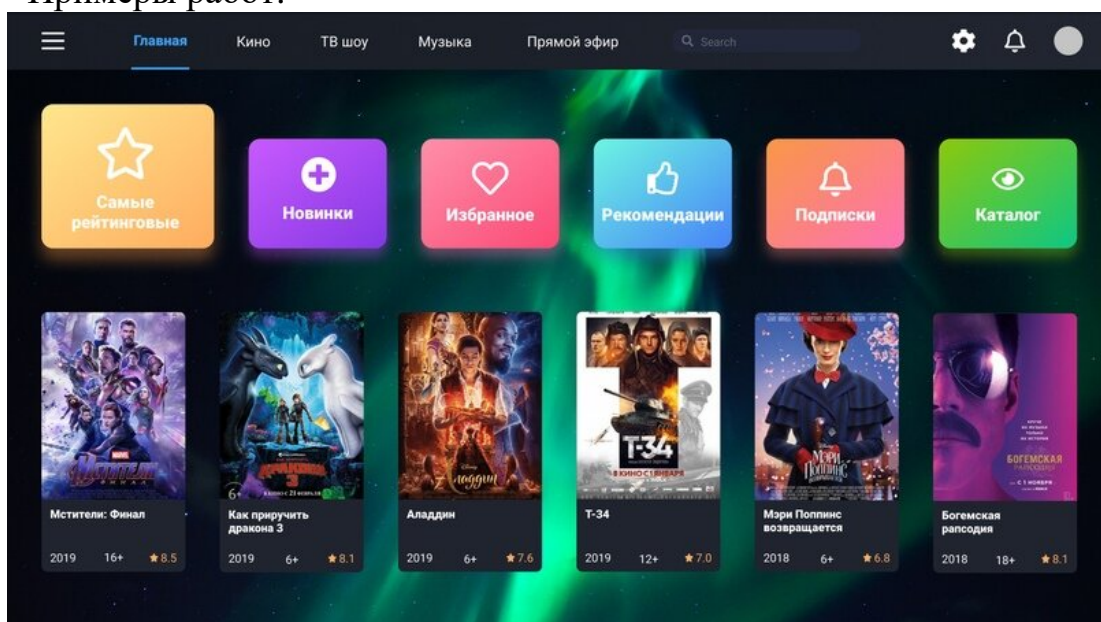
— умение принимать информированные и осмысленные решения по выбору библиотеки и технологии для решения проблемы.

Итог: Базовый курс по алгоритмам и структурам данных с практикой на JavaScript. Рассматриваются сложность алгоритмов, сортировка и т.д.

Верстка и программирование

HTML, CSS и JavaScript будет изучаться на 2 курсе, во второй половине в дисциплине «МДК 05.02 Разработка кода информационных систем». Это необходимая база, без которой войти в мир веб-разработки не получится. Изучаться языки будут на практике, работая над реальными проектами. Это позволит освоить сопутствующие инструменты: графические редакторы от Adobe (например, Illustrator/ XD / Photoshop) и Figma, редактор кода (в техникуме используется на начальных занятиях VS Code, а далее происходит переход на WebStorm), инструменты разработчика в браузерах.

Примеры работ:



1 из 4

Браузерные API. На 3 курсе, предположим, темой курсового проекта станет сервис стриминга музыки. Будет задача, чтобы трек останавливался, когда пользователь снимает Bluetooth-наушники. В JavaScript нет средств работы с Bluetooth, но разные браузеры предоставляют их в виде Application Programming Interface. Это интерфейсы для взаимодействия с чем-либо, например, с Bluetooth-устройствами. В браузерах есть и другие API: для определения геолокации пользователя или для взаимодействия с веб-камерой. Итог обучения на 3 курсе — разработка полноценного веб-сервиса.

На протяжении 2-4 курсов часто в проектах будут использоваться фреймворки. Они облегчают разработку больших проектов, делают их развитие и поддержку менее затратными. Многие фреймворки (например, React или Vue) позволяют думать об интерфейсе как о наборе

отдельных компонентов. Создав компонент однажды, его можно использовать в других местах сайта или даже в других проектах.

Взаимодействие с сервером. На 3 и 4 курсах потребуются знания и понимание, как работает HTTP-протокол и как устроено общение клиента и сервера. Это позволяет получать или отправлять данные и настроить взаимодействие со сторонним веб-сервисом, если это необходимо.

МДК.09.02 Оптимизация web-приложений рассказывает о SEO-продвижении проектов, работе в Яндекс.Директе / Google Analytics.

На протяжении всего обучения в техникуме будут развиваться личностные навыки. Веб-разработчику важно много общаться с дизайнерами, менеджерами и другими разработчиками. Еще пригодится умение работать с критикой: спокойно принимать замечания, признавать ошибки и исправлять их. Умение оценить задачу, распланировать время и выдержать сроки очень важно для этой профессии.

Где я встречу менеджеров и дизайнеров, если в техникуме нет таких специальностей?

На 2 курсе (во второй половине года) появится дисциплина «МДК 05.01 Проектирование и дизайн информационных систем».

В рамках этого МДК (междисциплинарного курса) будет происходить изучение круга задач менеджера проектов. Речь идет о грамотном проектировании проектов с точки зрения идей, инструментов и финансовой составляющей. Чтобы быть компетентным веб-разработчиком, нужно владеть базовыми смежными профессиями, например, менеджер проектов / веб-дизайнер / тестировщик / аналитик. Веб-дизайн будет рассмотрен лишь базово в этом МДК.

Более глубоко изучение разработки медиа-контента будет на МДК.08.02 Графический дизайн и мультимедиа. Разработка фирменного стиля, дизайн интернет-магазинов / сайтов / веб-сервисов / 2д-игр — это не единственное направление данного МДК. Обработка звука и создание небольших рекламных роликов с инфографикой — популярное требование к дизайнерам в ИТ — компании.

Зачем веб-разработчику владеть этим всем?

Без знания понятий в веб-дизайне и соответствующих программ, которых он разрабатывается — разработать проект не получится. Нужно уметь читать макеты дизайнеров и четко понимать, что требуется сделать. Умение делать медиа-контент включая ролики, постеры и т.д. — это дополнительный опыт в копилку при разработке проектов. Вы понимаете, как должен отображаться контент, какие есть подводные камни при верстке площадки под данный контент — вы сразу делаете с учетом всех требований.

ОП.01 Операционные системы и среды подразумевает обучение при работе с виртуальными машинами и развертыванием серверов на других машинах.

Любой веб-разработчик должен уметь пользоваться как хостингами, так и выделенными удаленными серверами на Linux. Там Вы будете учиться разворачивать свой проект.

Когда Вы научитесь пользоваться несколькими операционными системами, можно будет задуматься уже о тестировании Вашего проекта на других ОС (операционных системах). МДК.05.03 Тестирование информационных систем является ключевым предметом в данном вопросе. Проект будет тестироваться как в разных браузерах, так и ОС. Он будет прогоняться на защиту данных и т.д. Ведь Вы же не хотите, чтобы через день после релиза Ваш проект уже стал не Ваш?)

Если говорить более серьезно о веб-разработке, то на самом деле проекты, которые выходят на федеральный / мировой уровни разрабатываются по определенным критериям и правилам. ОП.09 Стандартизация, сертификация и техническое документирование именно про это. На любой проект необходимо разрабатывать документы. Например, техническое задание на интернет-магазин, инструкцию пользователя, тестирование системы и т.д. Есть принятие стандарты как в России (свои), так и в мире в целом. Чтобы проекты становились на уровень выше, нужно знать и уметь применять на практике требования к типу системы, которую Вы разрабатываете.

Как Вы думаете нужна ли математика веб-разработчикам? На этот вопрос Вам ответит дисциплина — ОП.10 Численные методы. Ну, а пока есть время — почитайте об алгоритмах машинного зрения ◉◉◉

После техникума (или во время обучения) я смогу устроиться на работу в ИТ — компанию?

Сколько времени потребуется — зависит от исходных позиций и промежуточных результатов. Тот, кто сейчас не знает ничего, но готов посвящать занятиям больше десяти часов в неделю, сможет приступить к поиску первых проектов примерно через полгода. Лучше начать оценивать рынок и присматриваться к вакансиям еще в процессе учебы. Это поможет детальнее познакомиться с требованиями работодателей и подтянуть нужные навыки.

Я постоянно буду учиться только в техникуме или будут какие-то дни работы в компаниях или на предприятиях?

На 3 и 4 курсах будут учебные и производственные практики.

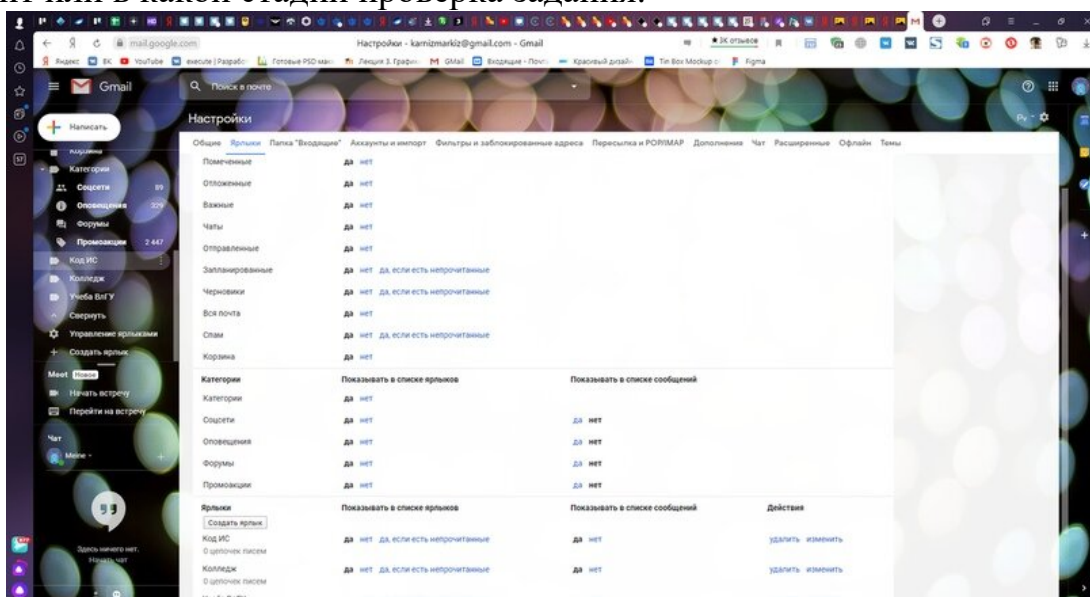
Учебная практика — это разработка проекта для техникума. Длится обычно 2-3 недели.

Производственная практика — это работа в компании в качестве стажировки или любого другого предприятия. Главное, чтобы выполняемая работа была связана с веб-разработкой.

Например, есть знакомая — работает в цветочном магазине, у них нет сайта. Итог: разработка интернет-магазина для этого индивидуального предприятия или компании, если успешно пройдет собеседование на позицию: стажер. Также производственную практику можно пройти и на базе мастерской «Веб-дизайн и разработка» и выполнить заказ от техникума, например, разработка веб-сервиса голосования для тематических мероприятий для "наименование" техникума".

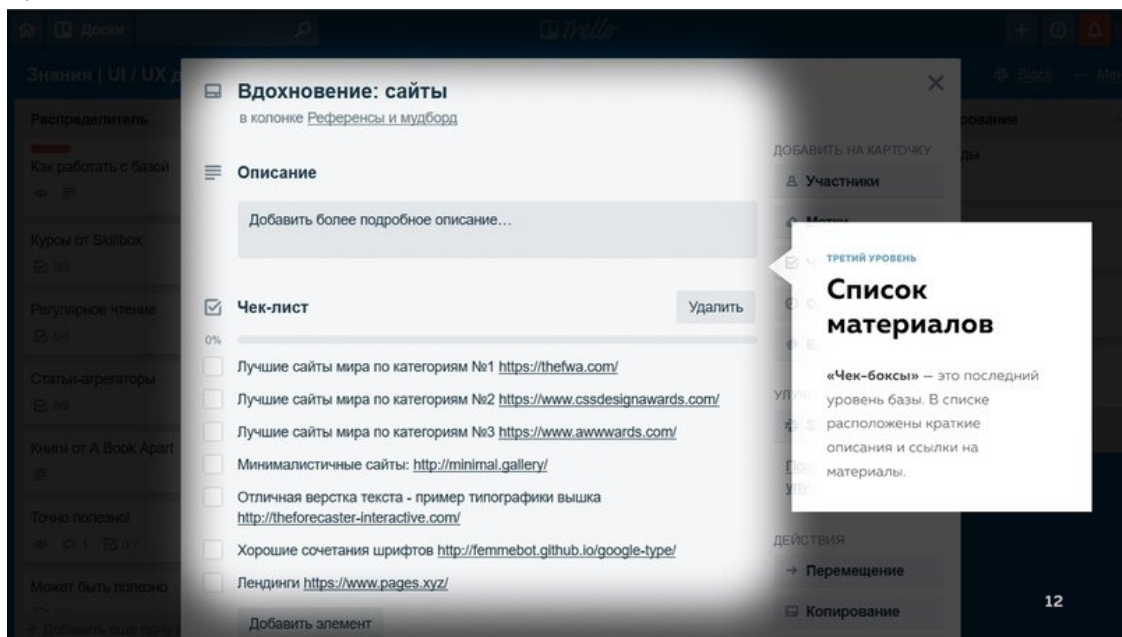


Организируйте систему тегов и папок, если общаетесь по электронной почте. Архивируйте завершённые переписки, например, когда задание сдано и доделок не требуется. Преподаватели могут создать папки по студентам, студенты — по курсам. Тегами можно отмечать, на какую оценку идёт студент или в какой стадии проверка задания.



Доска с задачами. Календарь показывает вам крайний срок сдачи задания, но вы можете установить свой: лучше сдать раньше и иметь больше времени на исправления. Заведите доску в Trello: так вы сможете видеть на одном экране, какие задания в процессе, а к каким вы ещё не приступили. Добавляйте в Trello не только сами задания, но и личные задачи. Например,

«разобраться с темой подробнее», «спросить у преподавателя про...» и прочее.



1 из 4

Попробуйте технику Pomodoro. Отключите на 25 минут всё, что может отвлечь, и погрузитесь в сфокусированную работу: чтение, письмо, решение задачи. Используйте обычные часы или romodoro-таймер. 25 минут — это стандартное время, в течение которого человек способен сфокусироваться, но вы можете адаптировать его под себя: сделать отрезок короче или длиннее. В конце сессии(25 минут, или того времени, что Вы установили) придумайте короткое вознаграждение: разомнитесь, выпейте чашку кофе, зайдите на любимый сайт в интернете.

Отрезки времени, на которые делится работа, условно называются помидорами. Лучший вариант: один «помидор» длится 30 минут: 25 минут работы и 5 минут отдыха.



Пользуйтесь разными ресурсами для обучения и саморазвития, не ограничивайтесь только тем материалом, что дают в техникуме. Не стесняйтесь задавать вопросы преподавателям во время или после лекций,

просите совета у одноклассников, если зашли в тупик, разбирайте задачи вместе, если для этого нет преград.

Развивайте не только навыки и умения программировать, но и человеческую часть сознания.

Зачем программисту soft skills?

Soft skills, «гибкие навыки», — это умение общаться, вести переговоры, организовывать работу, лоббировать, публично представлять технологию или продукт.

Необходимые веб-разработчику «гибкие навыки»:

- ответственность,
- умение работать в команде,
- эмоциональный интеллект,
- лидерские качества,
- логическое мышление,
- менеджмент.

Если не развивать себя как личность, то работать в сфере будет не так приятно как хотелось.

Например, успешно пройденное собеседование на этапе — программирование, может быть недостаточным из-за психологического настроя на антисоциальное поведение.

Сильные компании, в первую очередь — команда.

ИТ-сфера постоянно развивается, а точнее каждый день. Исходя из опыта прошлых лет, начитайте изучать основы уже сейчас, пока есть свободное лето. Аттестат лежит в одном из техникумов, копии направлены по запасным вариантам.

Рекомендации по летнему времяпрепровождению перед поступлением в техникум:

— Соберите информацию по будущей профессии, например, почитайте статьи на хабре, яндекс.академии, скиллбоксе и тд.

— Пройдите бесплатные мини-курсы на HTML- академии, Яндекс.Практикуме, Skillbox и тд.

— Скачайте бесплатные полные циклы курсов, если нужна помощь в поиске напишите мне.

— Прочитайте книги по менеджменту, психологии, бизнесу и алгоритмам. Предлагаем облако студентов веб-направления.

— Не останавливайтесь в развитии и постоянно учитесь. Читайте каждый день статьи на хабре, яндексе и тд. Делайте мини-проекты, тренировочные задания.

— Изучайте английский. Без знания технического английского — будет очень трудно. Сейчас в компаниях очень приветствуется и разговорный.

— Ищите работу, исходя из того, какой опыт вы можете там получить. Не исключен выход на фриланс - биржи.

— Разбирайтесь в инструментах, от которых зависит эффективность вашей работы. Просмотрите на ютубе обзоры основных инструментов для

ИТ-компаний. Поймите для чего они используются и научитесь применять для своих проектов / работы / подработки.

— Качайте soft skills. Постарайтесь подружиться с одногруппниками и выстройте свою команду современной ИТ -компании.

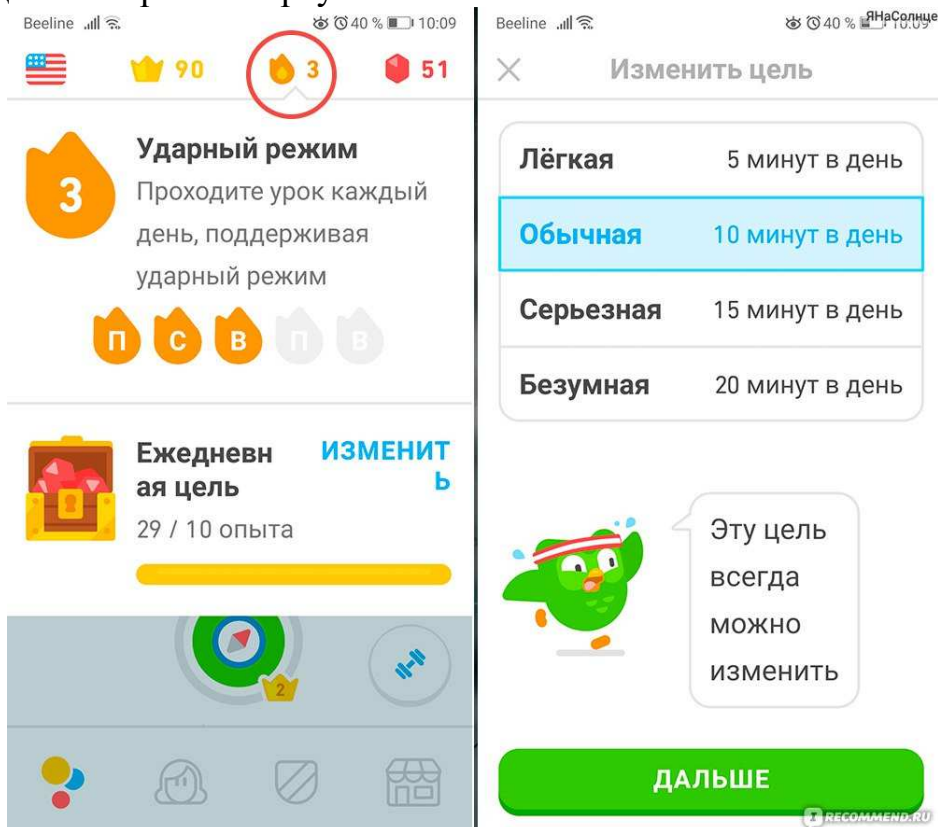
— Ставьте себе амбициозные цели. СПО учреждения каждый год участвуют в чемпионате WorldSkills Russia по компетенции «Веб-дизайн и разработка». Студенты победители часто успешно проходят собеседование и устраиваются в одну из ведущих компаний в области, например, сейчас в тренде компании, которые специализируются на аутсорсинге веб-разработки для бизнеса и digital-агентств.

— Интересуйтесь не только ИТ, но и другими сторонами жизни. Нужно уметь поддерживать темы для разговора и просто радоваться жизни (•_•)

Кстати о выборе иностранного языка.

Изучайте английский все время, каждый день. Вам будет в разы проще программировать, или при верстке придумывать классы.

Да и везде нужен английский на самом деле. Если учите или учили немецкий — срочно переучивайтесь.



Немного о предложениях техникума связанных с данной квалификацией

- Региональные этапы чемпионата «Молодые профессионалы (WorldSkills Russia)» по компетенции «Веб-дизайн и разработка». Итог: победитель обычно всегда успешно проходит собеседование в ИТ-компанию.