

СОВЕРШЕНСТВОВАНИЕ ОБРАЗОВАТЕЛЬНЫХ СТРАТЕГИЙ И МЕТОДИК ПРЕПОДАВАНИЯ ВЕБ-ДИЗАЙНА ДЛЯ ВЗРОСЛЫХ В УСЛОВИЯХ ЦИФРОВОЙ ТРАНСФОРМАЦИИ И ВНЕДРЕНИЯ ИСКУССТВЕННОГО ИНТЕЛЛЕКТА

Косак А.А., Полубок В.А.

Институт информационных технологий БГУИР, г. Минск

kosak@bsuir.by, polubok@bsuir.by

В статье рассматриваются проблемы организации дополнительного профессионального образования взрослых на примере специальности «Веб-дизайн и компьютерная графика». Анализируются ограничения традиционной университетской модели при обучении слушателей с высокой занятостью и жесткими карьерными сроками. Предлагаются современные стратегии и методики обучения. Особое внимание уделяется интеграции онлайн-среды и искусственного интеллекта (ИИ) в образовательный процесс как обязательного инструмента, повышающего мотивацию и конкурентоспособность выпускников.

Ключевые слова: дополнительное образование взрослых, веб-дизайн, компьютерная графика, проектное обучение, онлайн-технологии, искусственный интеллект.

У слушателя, пришедшего на переподготовку по специальности «Веб-дизайн и компьютерная графика, нет времени на долгие раскочки. У него за плечами другая профессия, работа, возможно семья, кредиты и жесткий дедлайн: «через год, может полтора я должен зарабатывать этим». Он не станет с умилением перерисовывать шрифтовые композиции ради искусства. Ему нужны навыки, которые сразу лягут в портфолио и конвертируются в заказы или предложение работодателя о трудоустройстве. И это на современном этапе меняет все. Традиционная модель – лектор вещает, слушатель слушает и конспектирует, потом лабораторная через неделю или две. Имея такую насыщенную жизнь за пределами университета, слушатель забывает 70% услышанного через час, если не применил сразу. Он теряет мотивацию, если не видит быстрого результата. Он раздражается, когда его заставляют учить синтаксис CSS-селекторов неделю, прежде чем показать, как сделать красивую кнопку. Следовательно, стратегии, методики и технологии должны создаваться под такого слушателя. А еще – под мир, где онлайн уже не дополнительная опция, а базовая среда, а искусственный интеллект стучится в каждую IDE и Figma-плагин. Не использовать ИИ в обучении сейчас – все равно что учить вождению на автомобиле без руля.

Главная ошибка программ дополнительного образования – они копируют университетский подход. Сначала теория, потом практика. У слушателя переподготовки на это нет времени. Нужен другой подход, например:

Слушатель дома смотрит короткие (10-15 минут) видео по конкретному инструменту: например, как работать с авто-лейаутами в Figma или как подключить шрифт через CSS. В классе (или онлайн) он не слушает лекцию, а сразу делает. Вместе с преподавателем разбирает реальный макет, решает проблему, почему, например, блок не центрируется. Это резко повышает плотность контакта с материалом.

Или – проектный поток, а не набор дисциплин. Вместо отдельных дисциплин «Верстка веб-страниц», «Компьютерная графика», «Проектирование динамических страниц» – сквозной проект, который идет от первого занятия до последнего. Сначала слушатель проектирует в Figma лендинг для вымышленной, но правдоподобной компании. Потом верстает его на HTML/CSS, делает адаптив. Потом добавляет интерактив на JS. Потом подключает к простому бэкенду, который сам же и написал. Так все темы оказываются нанизанными на один стержень. Ничего не забыто, ничто не висит в воздухе.

Или подход, где ошибка является частью обучения. Многие слушатели боятся ошибаться. Они стесняются спрашивать, потому что «я уже взрослый дядя, как можно не

знать ...?». Нужно создавать безопасную среду, где ошибка – не повод для стыда, а точка роста. Например, код-ревью в доброжелательном формате, где преподаватель говорит: не у тебя ошибка, а давай посмотрим, что здесь можно улучшить. Или обязательная практика парного программирования на занятиях.

Нужно менять методики преподавания. Первое, что нужно сделать – отказаться от лекций в чистом виде. Слушатель не способен слушать, например, про селекторы больше 15 минут. Поэтому нужно вводить микро-лекции на 10-15 минут, потом сразу задание на 30-40 минут. Это разбивка на мелкие осмысленные куски. Причем задания должны быть не «повторить за преподавателем», а с элементом исследования: «а что будет, если убрать `display: flex?`», «как сделать, чтобы блоки переносились на новую строку?».

Второе – это нужно дать возможность попробовать сделать самому, прежде чем узнаешь правильный способ. Например, прежде чем рассказывать про CSS Grid, попросить слушателей сверстать типовую сетку блога, используя только `float` и позиционирование. Они помучаются, поймут трудности, а потом, когда вы покажете Grid, все станет на свои места. Такое обучение называется проблемно-ориентированным, и оно работает для слушателей идеально.

Третье – это формирующее оценивание вместо экзаменов. Слушатель не очень хочет сдавать экзамен, например, по верстке. Он хочет знать, что его сайт действительно сверстан хорошо. Поэтому нужны регулярные мини-ревью, чеклисты самопроверки, или метод, когда студенты проверяют работы друг друга по заданным критериям. Итоговая оценка – не балл, а публичное портфолио на GitHub и в Figma, которое слушатель показывает потенциальному заказчику. Преподаватель лишь дает обратную связь: это хорошо, это доработать, а вот эту технику стоит освоить, чтобы ты мог претендовать на джуниор-позицию.

Особняком стоит методика работы с разнородной группой. На переподготовку приходят и бывшие бухгалтеры, и музыканты, и юристы. Уровень у некоторых нулевой, но мотивация разная. Кому-то нужна база, кому-то – сразу сложные фишки. Как быть в этой ситуации? Здесь можно использовать треки. Например, на одном занятии дается обязательное ядро для всех, а потом дополнительные задания на выбор: простой трек (доделать макет), средний (оптимизировать изображения), сложный (написать микро-анимацию на GSAP). Каждый идет по своему маршруту, но все в одном потоке.

Дополнительное образование взрослых сегодня – это в основном онлайн или гибрид. Но онлайн не равно вебинар в зумераз или два в неделю. Это целая экосистема, которую нужно проектировать.

Во-первых, нужна единая среда для всего: не мессенджер для чатов, не облако для файлов, не отдельная платформа для тестов. Идеально – что-то вроде Notion или Miro, где есть и доска с расписанием, и база знаний, и галерея работ, и чат. Слушателю не удобно прыгать между разными вкладками. Он хочет открыть одно место и увидеть: задание на неделю, ссылка на видео, форма сдачи, комментариев преподавателя.

Во-вторых, синхронные занятия должны быть интерактивными. Не 4 часа говорящей головы. А короткие блоки: 10 минут – демонстрация экрана, 15 минут – слушатели делают в своих средах (например, Figma), потом один показывает результат через шаринг экрана, разбор ошибок. Использовать опросы, доски Miro для совместного мозгового штурма. Без этого слушатель отключается через 20 минут.

В-третьих, асинхронные материалы – не только видео. Видео хорошо для демонстрации процесса (как нажать, куда перетащить). Но для теории – короткие текстовые конспекты с подсветкой кода, схемы, интерактивные тренажеры типа CodePen, где можно прямо в браузере править код и видеть результат. И обязательно – форум или чат, например, в Discord, где преподаватель отвечает в течение суток. Потому что, когда слушатель застрял в 21 час на верстке хедера, ему нужно решение сейчас, а не на следующем занятии.

Четвертое – запись всех занятий и доступ к ней навсегда. Слушатели пропускают по работе, по болезни, по семейным делам. Если нет записи – человек выпадает навсегда. Но запись не должна быть просто файлом. Она должна быть с оглавлением: на 15-й минуте говорим про flex, на 42-й – про адаптив». Тогда слушатель может перемотать к нужному месту, не пересматривая все подряд.

Самая горячая тема современности –искусственный интеллект. И это не футуризм, а реальность, которая уже изменила работу веб-дизайнера и разработчика. Если программа обучения игнорирует ИИ, она выпускает людей, которые будут неконкурентоспособны. Но использовать ИИ нужно с умом, иначе слушатели разучатся думать самостоятельно.

Первое направление – научить ИИ как инструменту. Показать слушателям, как с помощью Midjourney или DALL-E генерировать референсы для дизайна, создавать заглушки для иконок, подбирать цветовые палитры. Как через ChatGPT писать регулярные выражения, генерировать тестовый контент для макета, объяснять сложные темы простыми словами. Как использовать GitHubCopilot для автодополнения кода – но с четким правилом: не копируй вслепую, а разбирай каждую строчку, которую предложила нейросеть.

Второе направление – использование ИИ в роли тьютора. Слушателю часто нужна обратная связь «здесь и сейчас». Преподаватель не может отвечать 24/7. А вот чат-бот на базе API GPT, обученный на материалах курса, – может. Слушатель загружает свой HTML-код, бот проверяет его на валидность, указывает на ошибки в семантике, предлагает улучшения. Или слушатель описывает проблему (у меня не работает медиазапрос) – бот задает уточняющие вопросы и ведет к решению, не давая готового ответа сразу. Это развивает самостоятельность.

Третье направление – этика и границы. Важнейшая часть, которую нельзя пропустить. Слушатель должен понимать: ИИ может генерировать код, но не несет ответственности за его безопасность, доступность, производительность. Он может придумывать API, которых не существует. И самое главное – полагаться на ИИ без критического мышления – путь в никуда. Преподаватель должен устраивать специальные занятия: найди три ошибки в коде, который сгенерировал Copilot или сравни два решения – человека и ИИ, и объясни, почему сделанное человеком решение лучше».

Кроме того, нужно учить слушателей использовать ИИ для обучения самого преподавателя. Например, нейросеть может сгенерировать 10 вариантов практических заданий на тему адаптивной верстки разного уровня сложности. Может переформулировать сложную тему про каскадность несколькими способами для разных учебных стилей. Может анализировать ответы слушателей на форуме и подсвечивать типичные затруднения, чтобы преподаватель знал, что нужно разобрать на следующем занятии.

Но есть и риск. Если разрешить ИИ делать за слушателя всю работу, он никогда не научится. Поэтому правило простое: на начальных этапах ИИ запрещен. Слушатель должен освоить HTML, CSS, основы JS без подсказок. На среднем этапе – разрешен, но с условием, что он комментирует и объясняет каждую сгенерированную строчку. На продвинутом – добро пожаловать в реальный мир, где Copilot– твой помощник, но финальная ответственность за код твоя.

Нужно сделать обучение проектным и практическим с первого дня. Наши программы по специальности «Веб-дизайн и компьютерная графика»– это не каменная таблица. Ее можно и нужно менять, дополнять, ужимать, расширять. Главное – помнить, для кого мы это делаем. Для взрослых людей, у которых нет времени на раскачку. Они пришли к нам за навыками, которые изменят их жизнь. Нужно не подводить их.

Литература

1. Змеев, С.И. Андрагогика: основы теории, истории и технологии обучения взрослых: монография / С. И. Змеев. — Москва: ПЕР СЭ, Ай Пи Ар Медиа, 2024. – 272 с.

2. Минаков, А. И. Искусственный интеллект и нейросети в образовании: учебник / А.И. Минаков. – Москва: Директ-Медиа, 2024. – 164 с.
3. Тихомирова, Е. Обучение со смыслом: 13 правил для тех, кто учит взрослых / Е. Тихомирова. – Москва: Альпина ПРО, 2024. – 336 с.