

Проектирование интерактивного пианино на базе Web Audio API с использованием больших языковых моделей на примере Qwen Studio

Пайдершев Алексей Константинович

Приамурский государственный университет имени Шолом-Алейхема

Студент

Аннотация

В статье исследуются возможности проектирования интерактивных веб-приложений большими языковыми моделями на примере разработки многорежимного браузерного синтезатора в среде Qwen Studio. Выбор данного технологического стека, функционирующего на базе нативного интерфейса Web Audio API, обусловлен актуальным запросом на производительные мультимедийные решения, способные генерировать звук без предзагрузки тяжеловесных аудиофайлов и сторонних библиотек. Применение этого подхода позволяет существенно оптимизировать процессы создания сложного программного обеспечения, делегируя нейросети роль системного архитектора посредством структурированного промпт-инжиниринга. В то же время использование строгой математической установки параметров звука способствует объективной фиксации способности ИИ к выстраиванию сложной логики, глубокому структурированию архитектуры веб-приложения и достижению стабильной полифонии в условиях строгих браузерных ограничений.

Ключевые слова: Qwen Studio, нейросети, промпт-инжиниринг, Web Audio API, интерактивные веб-приложения.

Designing an interactive piano based on the Web Audio API using large language models using Qwen Studio as an example

Paydershev Alexey Konstantinovich

Sholom-Aleichem Priamursky State University

Student

Abstract

The article explores the possibilities of designing interactive web applications with large language models using the example of developing a multi-mode browser synthesizer in the Qwen Studio environment. The choice of this technology stack, which is based on the native Web Audio API interface, is driven by the current demand for high-performance multimedia solutions that can generate sound without preloading heavy audio files or third-party libraries. This approach significantly optimizes the process of creating complex software by delegating the role of a system architect to a neural network through structured reverse engineering. At the same time, the use of a strict mathematical setting of sound parameters contributes

to the objective fixation of AI's ability to build complex logic, deeply structure the architecture of a web application, and achieve stable polyphony under strict browser limitations.

Keywords: Qwen Studio, neural networks, prompt-engineering, Web Audio API, interactive web applications.

1 Введение

1.1 Актуальность

Сегодня создание веб-приложений с помощью ИИ вышло на совершенно новый уровень, когда он выполняет задачи полноценного Senior-архитектора, который сам пишет код программы, данное явление называется «вайб-кодинг». Пример с виртуальным пианино наглядно доказывает, что промпт-инжиниринг стал настоящим жёстким навыком. Чтобы получить качественный результат, недостаточно просто сказать «сделай синтезатор» – разработчик должен обладать предметными знаниями, понимая принципы работы осцилляторов и физику звука. Именно синергия глубокого понимания задачи и возможностей генеративного ИИ позволяет собирать сложные интерактивные микро-сервисы буквально за пять минут.

С технической точки зрения этот проект идеально отражает глобальный тренд на «чистый» и производительный веб без лишних зависимостей. Вместо того чтобы перегружать приложение десятками мегабайт скачанных аудио-сэмплов, современные подходы используют Web Audio API – технологию, которая генерирует звук напрямую в процессоре устройства с помощью математики. Это не только радикально ускоряет работу в браузере, но и максимально снижает порог входа в аудио-программирование. Теперь любой энтузиаст может создавать полноценные мультимедийные инструменты, не изучая сложные языки программирования и не подключая ни единой сторонней библиотеки.

1.2 Обзор исследования

С. В. Рогатин и В. А. Антошкин исследуют базовые принципы и специфику разработки браузерных компьютерных игр на основе технологии HTML5. Авторы анализируют инструментарий, позволяющий создавать кроссплатформенный мультимедийный контент без привлечения сторонних плагинов. И. А. Фёдоров и Е. Ю. Кузьменкова демонстрируют практическую реализацию игровых механик на примере разработки конкретного веб-приложения «League of football heads». В работе детально описывается процесс комплексного применения связки HTML5, CSS3 и JavaScript для выпуска полноценного интерактивного продукта. И. В. Елманов предлагает строгий инженерный подход к генерации кода с помощью искусственного интеллекта, рассматривая методологию Spec Driven Development для валидации процесса вайб-кодинга. Автор делает акцент на необходимости четкой спецификации и контроля качества при делегировании разработки нейросетям. А. А. Шинкарев, М. В. Ядрышникова, О. В. Логиновский, С. А. Лазарева и В. М. Губин проводят масштабный анализ эволюции применения

машинного обучения и ИИ в архитектуре корпоративных информационных систем. Исследование раскрывает, как интеллектуальные алгоритмы трансформируют механизмы работы современных систем поддержки принятия решений в бизнесе. Д. Р. Мухаметов рассматривает гуманитарный аспект применения нейросетей, исследуя концепцию «политического вайб-маркетинга». Работа посвящена анализу возможностей искусственного интеллекта в конструировании и продвижении привлекательного образа будущего в общественно-политическом пространстве.

1.3 Цель исследования

Цель исследования – наглядно продемонстрировать, как грамотный промпт-инжиниринг в сочетании с профильными знаниями позволяет делегировать нейросети роль Senior-разработчика при создании сложных веб-приложений без сторонних программ.

2 Материалы и методы

Методология исследования базируется на гибридном подходе, объединяющем структурированный промпт-инжиниринг и отказ от традиционных методов веб-разработки в пользу процедурной генерации. На первом этапе формируются строгие архитектурные ограничения: вместо использования предзагруженных аудиофайлов и тяжелых библиотек утверждается применение исключительно встроенного Web Audio API. Затем эти требования формируются в детализированную инструкцию для нейросети. Применяется метод ролевого позиционирования, а также жестко задаются математические и физические параметры звука: типы волн и настройки ADSR-ограничивающей для точной имитации акустических свойств четырех различных инструментов. В качестве платформы для написания кода применяется среда Qwen Studio.

3 Результаты

Первый шаг исследования – написание специализированного промта для нейросети, который содержит в себе параметры дизайна, генерации звука, управления и анимации.

Промт: «Выступи в роли Senior Frontend-разработчика. Напиши код для создания полностью рабочего интерактивного виртуального пианино с переключаемыми режимами звучания. Используй HTML. Код должен быть чистым, современным и работать в браузере без использования сторонних библиотек.

Технические и визуальные требования:

1. Дизайн и верстка:
 - Создай клавиатуру пианино минимум на одну полную октаву (от ноты «До» до ноты «Си»).
 - Черные клавиши должны находиться поверх белых и располагаться в правильных интервалах.

- Дизайн должен быть аккуратным: тени для объема, скругления углов.
 - Над клавиатурой создай панель управления с кнопками или выпадающим списком для выбора режима звучания: «Пианино», «Орган», «Синтезатор», «Электро».
 - 2. Генерация звука:
 - НЕ используй внешние аудиофайлы. Синтезируй звук с помощью
 - Настрой точные частоты нот (например, A4 = 440 Hz).
 - Реализуй 4 режима звучания, меняя тип волны осциллятора и параметры затухания в GainNode:
 - Пианино: Волна triangle или sine. Быстрая атака, плавное затухание.
 - Орган: Несколько осцилляторов. Нет затухания, пока клавиша нажата, резкое прерывание при отпускании.
 - Синтезатор: Волна sawtooth. Добавь легкий эффект хоруса и небольшое плавное затухание.
 - Электро: Волна sine или square. Быстрая атака с коротким ярким щелчком, затем ровный, чистый звук с долгим затуханием.
 - 3. Взаимодействие и управление:
 - Инструмент должен реагировать на клики.
 - Добавь поддержку клавиатуры компьютера (нижний ряд A-J для белых, W-U для черных).
 - 4. Анимация:
 - Визуальное нажатие клавиш при игре мышкой или с клавиатуры. Возврат в исходное состояние при отпускании.
 - Индикация активного режима на панели управления (выделенная кнопка).
- Выведи весь код в одном блоке, чтобы его можно было сразу запустить в браузере.»

Сгенерированный код доступен по ссылке:
https://drive.google.com/file/d/1U1TlIrVsrm8EYHWeItlDxz0_1_TiqNMPt/view?usp=sharing

Чтобы иметь к приложению постоянный доступ, необходимо создать ярлык для запуска на рабочем столе. Для этого нужно создать текстовый документ и скопировать в него сгенерированный код, а затем изменить формат файла на HTML (рис.1-2).

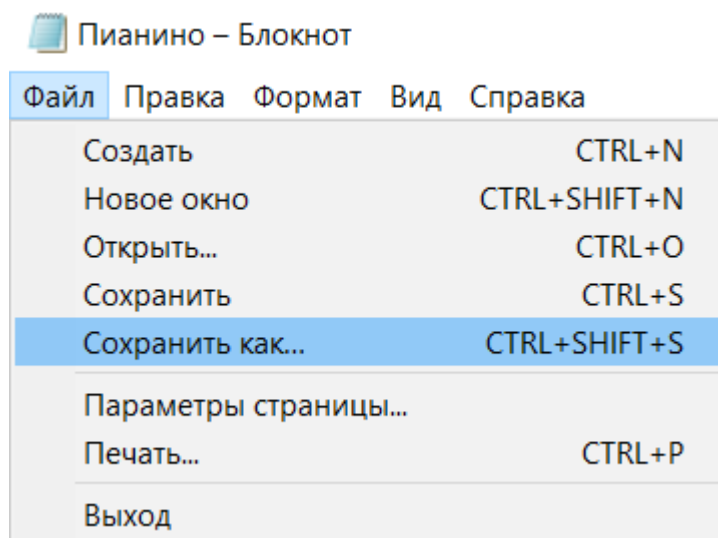


Рисунок 1. Сохранение файла

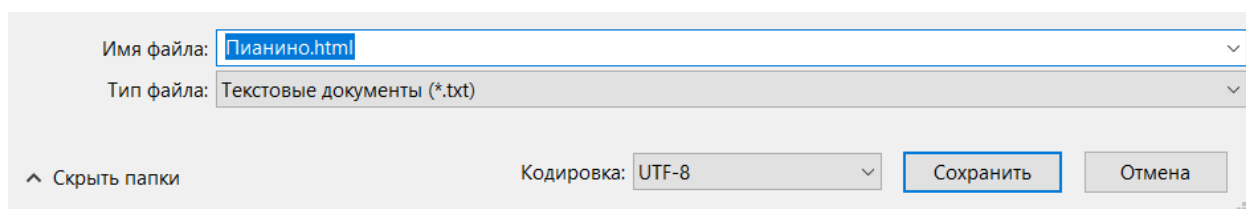


Рисунок 2. Сохранение файла

Теперь приложение запускается прямо с рабочего стола.

В результате применения описанной методологии успешно сгенерирован полностью рабочий, интерактивный веб-синтезатор. Весь проект уместился в один легковесный файл Пианино.html, не требующий установки прм-пакетов, поднятия локального сервера или скачивания сторонних медиафайлов. А само приложение имеет интуитивно понятный и эстетичный интерфейс (рис. 3).

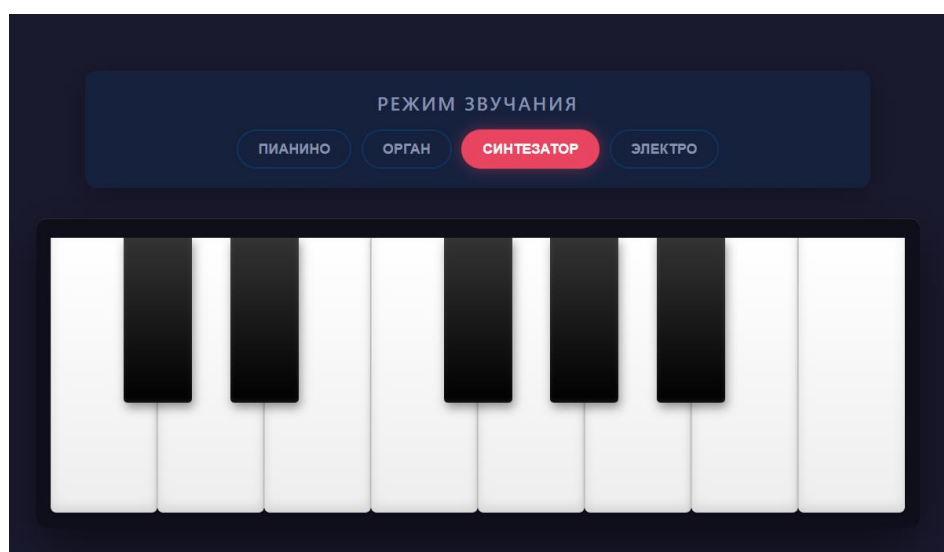


Рисунок 3. Интерфейс приложения

ИИ безупречно справился с задачей создания инструмента на чистом стеке. Использование Web Audio API доказало, что браузер способен аппаратно синтезировать звук в реальном времени, избавляя разработчиков от необходимости управлять тяжеловесными директориями с «.mp3» или «.wav» сэмплами.

Благодаря точным математическим инструкциям в промпте, нейросеть успешно реализовала четыре совершенно разных тембра. Изменение типов волн и ювелирная настройка фаз атаки и затухания позволили получить как бесконечно тянущийся звук органа, так и резкое, электронное звучание ретро-синтезатора 80-х (рис 4).

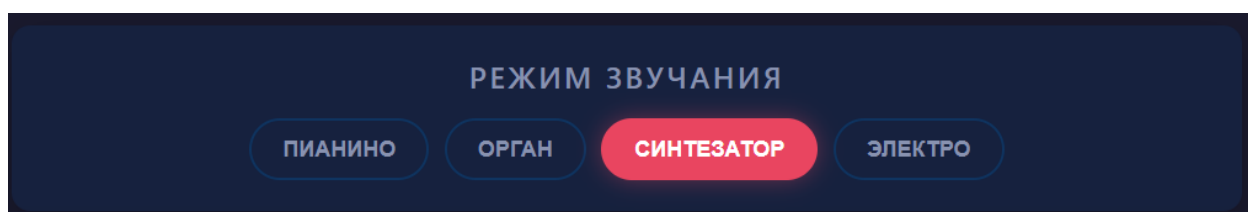


Рисунок 4. Переключение режимов

Скрипт корректно обрабатывает одновременные нажатия множества клавиш. Под каждую ноту динамически создается и уничтожается свой OscillatorNode, что исключает утечки памяти и обеспечивает плавное воспроизведение сложных аккордов без задержек и нагрузки на процессор (рис. 5).

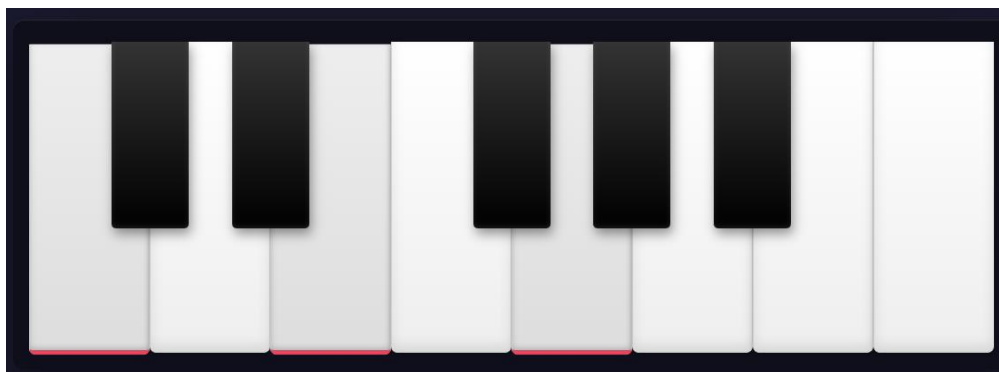


Рисунок 5. Нажатие нескольких клавиш одновременно

Результаты исследования говорят о том, что парадигма программирования сместилась. ИИ забирает на себя рутинное написание синтаксиса, но требует взамен сильного системного мышления. Человек в этом тандеме выступает не как кодер, а как Senior-архитектор и звукорежиссер, диктующий физические и логические законы системы.

4 Вывод

В заключение стоит отметить, что эксперимент по созданию многорежимного браузерного синтезатора выходит далеко за рамки простой демонстрации возможностей современных нейросетей. Он наглядно

иллюстрирует наступление новой эпохи в IT-индустрии, где скорость и качество разработки сложных интерактивных продуктов зависят не от навыка ручного написания строк кода, а от глубины понимания предметной области и владения промпт-инжинирингом. Использование нативных, ресурсоэффективных технологий, таких как Web Audio API, в связке с искусственным интеллектом открывает совершенно новые горизонты. Разработчик окончательно переходит в позицию визионера и системного архитектора: он формулирует концепцию, задает физические ограничения и математические параметры, полностью доверяя рутинную реализацию ИИ. Данный проект служит убедительным доказательством того, что синергия глубоких технических знаний и грамотно выстроенного диалога с нейросетью позволяет превращать смелые идеи в готовые, независимые и производительные приложения буквально за несколько минут.

Библиографический список

1. Рогатин С. В., Антошкин В. А. Разработка компьютерных игр по технологии html5 //Информатика и прикладная математика. 2015. №. 21. С. 12-16. URL: <https://elibrary.ru/item.asp?id=25030492>
2. Фёдоров И. А., Кузьменкова Е. Ю. Разработка игрового приложения «league of football heads» с использованием html5, css3, javascript //Новые математические методы и компьютерные технологии в проектировании, производстве и научных исследованиях. 2018. С. 311-311. URL: <https://elibrary.ru/item.asp?id=44778935>
3. Елманов И. В. Инженерный подход к ии: использование spec driven development для валидации vibe coding //Вестник науки. 2025. Т. 4. №. 12 (93). – С. 716-723. URL: <https://cyberleninka.ru/article/n/inzhenernyy-podhod-k-ii-ispolzovanie-spec-driven-development-dlya-validatsii-vibe-coding>
4. Шинкарев А. А., Ядрышникова М. В., Логиновский О. В., Лазарева С. А., Губин В. М. Эволюция использования и применения машинного обучения и искусственного интеллекта в разработке корпоративных информационных систем и в системах поддержки принятия решений //Вестник Южно-Уральского государственного университета. Серия: Компьютерные технологии, управление, радиоэлектроника. 2025. Т. 25. №. 4. С. 17-42. URL: <https://cyberleninka.ru/article/n/evolyutsiya-ispolzovaniya-i-primeneniya-mashinnogo-obucheniya-i-iskusstvennogo-intellekta-v-razrabotke-korporativnyh>
5. Мухаметов Д. Р. Политический вайб-маркетинг: возможности искусственного интеллекта для продвижения образа будущего //Известия Тульского государственного университета. Гуманитарные науки. 2025. №. 3. С. 27-33. URL: <https://cyberleninka.ru/article/n/politicheskiy-vayb-marketing-vozmozhnosti-iskusstvennogo-intellekta-dlya-prodvizheniya-obraza-buduschego>