

3D-ВИЗУАЛИЗАЦИЯ ОБЪЕКТОВ В ВЕБ-ПРИЛОЖЕНИИ С ИСПОЛЬЗОВАНИЕМ THREE.JS

Современные веб-технологии для трёхмерной графики становятся всё более актуальными благодаря расширению возможностей браузеров и росту интерактивного контента. Использование 3D-визуализации востребовано при создании виртуальных экскурсий по архитектурным объектам, демонстрационных моделей в образовании и интерактивных показов товаров в электронной коммерции. Целью работы является разработка прототипа веб-приложения для отображения и взаимодействия с 3D-объектами в браузере с использованием библиотеки Three.js.

В качестве основы был выбран движок Three.js - JavaScript-библиотека, предназначенная для рендеринга 3D-графики в браузере. Архитектура приложения базируется на использовании сцены, камеры и рендерера. Сцена (Scene) выступает контейнером для объектов, камер и источников света. Камера (Camera) определяет точку обзора на сцену, а рендерер (Renderer) отвечает за отрисовку, используя WebGL – низкоуровневый графический API. Управление пользователем обеспечивалось через модуль OrbitControls, позволяющий осуществлять вращение и масштабирование объекта на сцене. В ходе реализации были выполнены основные этапы настройки сцены и камеры, загрузки и отображения 3D-моделей, а также подключено управление через OrbitControls. Интерфейс был адаптирован под различные разрешения экранов, что позволило добиться корректного отображения в популярных браузерах. Проведённое тестирование подтвердило стабильную работу приложения в средах Chrome, Firefox, Edge и Safari. Управление камерой осуществляется плавно, геометрия моделей отображается корректно, текстуры загружаются без искажений.

Тем не менее, в процессе разработки были выявлены направления, требующие дальнейшей проработки. В частности, необходимо оптимизировать производительность при работе с моделями, содержащими свыше 500 000 полигонов. Также интерес представляет задача по поддержке анимации, в том числе скелетной, и реализации пользовательских аннотаций: измерений, комментариев и отображения метаданных. Одним из ключевых векторов развития является интеграция приложения с облачными хранилищами и бэкенд-API, что позволит обеспечить динамическую подгрузку контента и расширит функциональные возможности системы.