

**МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ
ДЕРЖАВНИЙ УНІВЕРСИТЕТ «КИЇВСЬКИЙ АВІАЦІЙНИЙ ІНСТИТУТ»
ФАКУЛЬТЕТ АРХІТЕКТУРИ, БУДІВНИЦТВА ТА ДИЗАЙНУ**

Кафедра комп'ютерних мультимедійних технологій

ДОПУСТИТИ ДО ЗАХИСТУ
Завідувач кафедри
_____ О.А. Бобарчук
« ____ » _____ 2025 р.

КВАЛІФІКАЦІЙНА РОБОТА

(ПОЯСНЮВАЛЬНА ЗАПИСКА)

ЗДОБУВАЧА ОСВІТНЬОГО СТУПЕНЯ “МАГІСТР”

**Тема: «Методи та засоби створення веб-платформи для музею техніки
магнітного запису з використанням інтерактивних 3D моделей»**

Виконавець: _____ Олексій ФІЛІПАС

Керівник: _____ к.т.н., доцент Олександр БОБАРЧУК

Нормоконтролер: _____ Світлана ГАЛЬЧЕНКО

КИЇВ 2025

ДЕРЖАВНИЙ УНІВЕРСИТЕТ «КИЇВСЬКИЙ АВІАЦІЙНИЙ ІНСТИТУТ»

Факультет архітектури, будівництва та дизайну

Кафедра комп'ютерних мультимедійних технологій

Спеціальність 186 Видавництво та поліграфія

Освітньо-професійна програма Технології електронних мультимедійних видань

ЗАТВЕРДЖУЮ

Завідувач кафедри

О.А. Бобарчук

« » 2025 р.

ЗАВДАННЯ

на виконання кваліфікаційної роботи

Філіпаса Олексія Олеговича

(прізвище, ім'я, по батькові здобувача вищої освіти в родовому відмінку)

1. Тема роботи «Методи та засоби створення веб-платформи для музею техніки магнітного запису з використанням інтерактивних 3D моделей»
затверджена наказом ректора від «23» серпня 2025 р. № 1496/ст.

2. Термін виконання роботи: з 26.08.2025 р. по 03.12.2025 р.

3. Вихідні дані до роботи: план створення інтерактивних 3D моделей та публікації їх у веб, текстові та графічні матеріали.

4. Зміст пояснювальної записки: Теоретико-методологічні основи інтерактивних музейних експозицій. Аналіз та проектування технологічних засобів веб-платформи. Створення інтерактивної моделі «Маяк-140» та оцінка ефективності процесу.

5. Перелік обов'язкового графічного (ілюстративного) матеріалу:

Презентаційний матеріал, скріншоти робочого процесу створення 3D, веб сторінки.

6. Календарний план-графік

№ пор.	Завдання	Термін виконання	Підпис керівника
1.	Ознайомлення з предметною областю та формулювання гіпотези. Підготовка референсних матеріалів та дослідження теми візуалізації.	02.10-07.10	
2.	Проведення аналізу науково-методичної літератури. Дослідження теорії мультимедіа та когнітивного навантаження.	07.10-20.10	
3.	Аналіз та обґрунтування методів візуалізації та інтерактивності.	01.11-07.11	
4.	Огляд технологічних засобів та бібліотек.	08.11-11.11	
5.	Підготовка 3D-моделі складного процесу в Blender.	12.11-20.11	
6.	Імплементация інтерактивної візуалізації.	21.11-25.11	
7.	Реалізація механізму інтерактивності та зворотного зв'язку.	26.11-30.11	
8.	Функціональне тестування та налагодження платформи.	01.12-05.12	
9.	Підготовка презентаційного матеріалу.	06.12-20.12	

7. Дата видачі завдання: «26» серпня 2025 р.

Керівник кваліфікаційної роботи _____
(підпис керівника)

Олександр БОБАРЧУК
(П.І.Б.)

Завдання прийняв до виконання _____
(підпис здобувача вищої освіти)

Олексій ФІЛІПАС
(П.І.Б.)

РЕФЕРАТ

Пояснювальна записка до кваліфікаційної роботи «Методи та засоби створення веб-платформи для музею техніки магнітного запису з використанням інтерактивних 3D-моделей» містить 87 сторінок, 10 рисунків, 2 таблиці та 48 використаних джерел.

ВЕБ-ПЛАТФОРМА, МУЛЬТИМЕДІА, ІНТЕРАКТИВНІСТЬ, МУЗЕЙНА ЕКСПОЗИЦІЯ, 3D-МОДЕЛЮВАННЯ, THREE.JS, WEB AUDIO API, GLTF/GLB, UX, ЦИФРОВА СПАДЩИНА, МАГНІТОФОН «МАЯК-140».

Об'єкт дослідження – цифрові музейні експозиції з інтерактивними 3D моделями.

Предмет дослідження – методи та технічні засоби створення веб-платформи з інтерактивними 3D-моделями для візуалізації технічних процесів магнітного запису.

Мета роботи – обґрунтувати й реалізувати методи створення веб-платформи, що поєднує інтерактивну 3D-модель магнітофону «Маяк-140» із засобами аудіовізуалізації для пояснення принципів магнітного запису та відтворення.

Методи дослідження: Теоретичні: аналіз літератури з мультимедіа, музейної педагогіки, когнітивних аспектів сприйняття; систематизація підходів до інтерактивних експозицій; огляд технологій 3D-візуалізації та веб-розробки. Практичні: 3D-моделювання (Blender), оптимізація сітки та текстур, проектування UI/UX, програмна реалізація інтерактивності з використанням Three.js та Web Audio API, створення механізмів синхронізації 3D-анімацій з аудіосигналом. Емпіричні: юзабіліті-тестування, оцінка освітньої ефективності та аналіз взаємодії користувачів із прототипом.

Практичне значення кваліфікаційної роботи полягає у можливості використання розробленого інтерактивного 3D-експонату в музеях технічної спадщини, освітніх установах та онлайн-експозиціях. Платформа може застосовуватися для популяризації техніки магнітного запису, проведення демонстрацій та підвищення мотивації користувачів до вивчення технічних процесів.

ЗМІСТ

ВСТУП.....	10
РОЗДІЛ 1. ТЕОРЕТИКО-МЕТОДОЛОГІЧНІ ОСНОВИ ІНТЕРАКТИВНИХ МУЗЕЙНИХ ЕКСПОЗИЦІЙ	13
1.1. Концептуальні засади мультимедіа.....	13
1.1.1. Сутність, компоненти та дидактичні функції мультимедіа	17
1.1.2. Психолого-педагогічні основи мультимедійного навчання	19
1.1.3. Візуалізація як ключовий компонент мультимедіа	21
1.2. Інтерактивність, проблемне навчання та музейна педагогіка.....	23
1.2.1. Інтерактивність як спосіб залучення та формування проблемного поля.....	24
1.2.2. Музейна педагогіка в епоху цифрових технологій та концепція «Digital Heritage».....	25
1.2.3. Когнітивний аспект та досвід користувача (UX) у музейному середовищі	26
1.3. Аналіз складних технічних процесів (на прикладі магнітного запису)..	27
1.3.1. Фізичні та технічні принципи магнітного запису та відтворення (як основа для моделювання).....	28
1.3.2. Обґрунтування використання 3D-візуалізації для демонстрації невидимих процесів	30
1.3.3. Вимоги до юзабіліті та доступності інтерактивних технічних експонатів.....	32
1.4. 1.4. Методології створення інтерактивних навчальних експонатів	33
Висновки до розділу 1	34
РОЗДІЛ 2. АНАЛІЗ ТА ПРОЄКТУВАННЯ ТЕХНОЛОГІЧНИХ ЗАСОБІВ ВЕБ-ПЛАТФОРМИ	36
2.1. Технологічний стек для створення та відтворення 3D-моделі.....	36

2.1.1. Порівняння інструментів 3D-моделювання та обґрунтування вибору Blender	37
2.1.2. Технології веб-візуалізації: WebGL та бібліотеки на його основі.....	39
2.1.3. Формати обміну 3D-даними та вибір glTF/GLB	40
2.2. Інтерактивність у цифровому музейному експонаті.....	42
2.2.1. Огляд підходів до інтерактивності та вибір оптимальної технології (Unity, Unreal Engine, Babylon.js, Three.js)	43
2.2.2. Three.js як інструмент взаємодії з моделлю	44
2.2.3. Реалізація аудіоінтерактивності за допомогою Web Audio API.....	46
2.3. Інтеграція 3D-моделі у вебплатформу.....	47
2.3.1. Архітектура вебдодатка та обґрунтування вибору SPA.....	47
2.3.2. Взаємодія компонентів: 3D-модель, UI та аудіопроцесор	48
2.3.3. Оптимізація продуктивності у браузері.....	50
2.4. Методика створення 3D-моделі магнітофона «Маяк-140»	51
2.4.1. Декомпозиція пристрою та моделювання ключових вузлів.....	52
2.4.2. Підготовка матеріалів і освітлення для PBR-рендерингу	53
2.4.3. Оптимізація й експорт моделі у форматі GLB	54
2.5. Проектування користувацького інтерфейсу (UI/UX).....	56
2.5.1. Принципи дизайну для навчальних музейних платформ	57
2.5.2. Забезпечення доступності вебплатформи (WCAG 2.1).....	58
2.5.3. Сценарії взаємодії користувача	59
Висновки до розділу 2	61
РОЗДІЛ 3. СТВОРЕННЯ ІНТЕРАКТИВНОЇ МОДЕЛІ «МАЯК-140» ТА ОЦІНКА ЕФЕКТИВНОСТІ ПРОЦЕСУ	63
3.1. Підготовка та імплементація оптимізованої 3D-моделі	63
3.1.1. Створення 3D-моделі магнітофона «Маяк-140» у Blender	64
3.1.2. Текстурування.....	65

3.1.3. Анімація за допомогою NLA, оптимізація моделі та експорт у GLB	67
3.1.4. Завантаження GLB-моделі у Three.js та ініціалізація сцени.....	71
3.1.5. Підготовка структури вузлів до інтерактивності.....	72
3.1.6. Тестування коректності геометрії, pivot-точок і анімацій після експорту.....	73
3.2. Створення інтерактивної логіки та аудіовізуальних модулів	74
3.2.1. Програмування взаємодії з моделлю у Three.js	74
3.2.2. Інтеграція Web Audio API для керування анімаціями та індикаторами	75
3.2.3. Реалізація динамічного VFD-індикатора та анімації механіки	76
3.3. Тестування та оцінка ефективності інтерактивної експозиції	77
3.3.1. Юзабіліті-тестування інтерфейсу та взаємодій.....	77
3.3.2. Оцінка освітньої ефективності цифрового експоната	78
3.3.3. Аналіз результатів та підтвердження практичної цінності вебплатформи.....	79
ВИСНОВКИ.....	80
СПИСОК БІБЛІОГРАФІЧНИХ ПОСИЛАНЬ ВИКОРИСТАНИХ ДЖЕРЕЛ.....	82
ДОДАТКИ.....	87

ПЕРЕЛІК УМОВНИХ ПОЗНАЧЕНЬ, СКОРОЧЕНЬ, ТЕРМІНІВ

3D-моделивання	Процес створення тривимірного представлення будь-якого об'єкта за допомогою спеціалізованого програмного забезпечення.
API (Application Programming Interface)	Набір визначень та протоколів, що дозволяє одному програмному продукту взаємодіяти з іншим.
Digital Heritage (Цифрова спадщина)	Сукупність цифрових ресурсів, що мають культурну, історичну або наукову цінність і потребують збереження.
glTF (GL Transmission Format)	Відкритий формат файлів для ефективноної передачі та завантаження 3D-сцен і моделей у веб-додатках.
PBR (Physically Based Rendering)	Метод рендерингу, який моделює взаємодію світла з поверхнями на основі фізичних законів для досягнення фотореалізму.
SPA (Single Page Application)	Веб-додаток або сайт, що завантажує єдину HTML-сторінку та динамічно оновлює її вміст під час взаємодії з користувачем.
Three.js	Кросбраузерна JavaScript-бібліотека, що використовується для створення та відображення анімованої 3D-графіки у веб-браузері.
UI (User Interface)	Інтерфейс користувача; сукупність засобів, за допомогою яких людина взаємодіє з програмою або пристроєм.

UX (User Experience)	Досвід користувача; сукупність вражень, які отримує користувач від взаємодії з продуктом.
VFD (Vacuum Fluorescent Display)	Вакуумно-люмінесцентний індикатор; тип дисплея, що використовувався в аудіотехніці для відображення рівня сигналу.
Web Audio API	Високорівневий програмний інтерфейс JavaScript для обробки та синтезу аудіо у веб-додатках.
WebGL (Web Graphics Library)	Програмна бібліотека для JavaScript, що дозволяє створювати інтерактивну 3D-графіку в межах сумісного веб-браузера.
Веб-доступність (Web Accessibility)	Властивість веб-ресурсу, що забезпечує можливість його використання людьми з різними фізичними та технічними можливостями.
Веб-платформа	Програмний комплекс, що працює у веб-середовищі та забезпечує інтерактивну взаємодію користувача з контентом через браузер.
Гейміфікація	Використання ігрових елементів і механік у неігровому контексті (наприклад, у навчанні) для підвищення залученості користувачів.
Еквалайзер	Пристрій або програмний модуль для корекції амплітуди звукового сигналу в різних частотних діапазонах.
Інтерактивність	Здатність інформаційно-комунікаційної системи активно реагувати на дії користувача.

ВСТУП

Актуальність теми. Розвиток цифрових технологій істотно вплинув на те, як музеї працюють із технічною та культурною спадщиною. Стає очевидним, що статичні експозиції вже не здатні повною мірою передати принципи роботи складних технічних систем, особливо коли йдеться про механізми, приховані всередині корпусу пристрою. Саме тому дедалі більшої уваги набувають мультимедійні формати, які дозволяють не лише показати об'єкт, а й пояснити його функціонування. Особливо актуальним є застосування інтерактивних 3D-моделей, оскільки вони дають можливість користувачеві самостійно досліджувати об'єкт, змінювати ракурси та бачити процеси, що зазвичай лишаються непомітними.

У випадку техніки магнітного запису це питання стоїть особливо гостро: магнітофони мають складну кінематику, а принципи запису та відтворення звуку є малоочевидними для широкої аудиторії. Фізична експозиція не здатна демонструвати внутрішні вузли в динаміці, а доступ до робочих механізмів часто обмежений. Тому створення веб-платформи з інтерактивною 3D-моделлю магнітофону «Маяк-140» відкриває можливість подолати ці бар'єри й подати технічний об'єкт у сучасному, пізнавальному форматі.

Особливого значення цьому дослідженню надає вибір самого об'єкта моделювання. Магнітофон «Маяк-140» - це не лише технічний пристрій, а й знаковий артефакт української індустріальної спадщини. Він був розроблений і виготовлявся на Київському заводі «Маяк» – одному з флагманів аудіоіндустрії Східної Європи. Продукція заводу формувала музичну культуру кількох поколінь, але сьогодні такі пристрої поступово зникають із побуту й стають малодоступними для дослідження. Тому створення інтерактивної 3D-моделі «Маяка-140» можна розглядати як цифрове збереження культурної пам'яті (Digital Heritage). Це дозволяє зафіксувати досягнення української інженерної школи та зробити їх доступними для вивчення у сучасному веб-просторі, інтегруючи локальну технічну історію в глобальний цифровий музей. Саме це визначає актуальність теми.

Мета дослідження полягає у визначенні та обґрунтуванні методів і засобів створення веб-платформи для музейної експозиції техніки магнітного запису, яка використовує інтерактивні 3D-моделі для пояснення складних технічних процесів.

Для досягнення мети були сформульовані такі **завдання**:

- проаналізувати принципи мультимедійної візуалізації та їхнє застосування в музейних експозиціях;
- дослідити інтерактивність і UX як чинники залучення користувачів;
- розкрити фізичні та технічні основи магнітного запису, визначивши процеси, які варто відтворити у 3D;
- обґрунтувати вибір технологічного стеку для 3D-графіки й аудіовізуалізації;
- спроектувати архітектуру веб-платформи та інтерфейс взаємодії;
- створити та оптимізувати 3D-модель магнітофону «Маяк-140» для інтерактивної інтеграції у веб-середовище;
- реалізувати механізми керування, синхронізацію анімацій із аудіосигналом та інші функції;
- провести тестування й оцінити ефективність створеного експонату.

Об'єкт дослідження – цифрові музейні експозиції з інтерактивними 3D моделями.

Предмет дослідження – методи та технічні засоби створення веб-платформи з інтерактивними 3D-моделями для візуалізації технічних процесів магнітного запису.

Для реалізації поставлених завдань застосовано комплекс **методів дослідження**. Теоретичні методи (аналіз, порівняння, узагальнення) використані для опрацювання джерел із мультимедіа, музейної педагогіки та візуалізації складних технічних процесів. Практичні методи включали 3D-моделювання, оптимізацію сітки, проектування UI/UX та програмну реалізацію інтерактивності за допомогою Three.js і Web Audio API. Емпіричні методи були використані для тестування інтерфейсу й оцінювання того, як користувачі взаємодіють із моделлю та наскільки зрозумілими стають принципи магнітного запису.

Наукова новизна полягає у створенні універсальної методології, яка охоплює процес розробки та інтеграції інтерактивної 3D моделі у веб-середовище.

Практичне значення полягає у можливості використання розробленої веб-платформи як частини музейної експозиції, освітнього онлайн-курсу або демонстраційного матеріалу для вивчення принципів магнітного запису. Ресурс може стати основою для подальших інтерактивних проєктів, присвячених технічній спадщині. Також важливо зазначити, що розроблена архітектура є універсальною і може бути масштабована для інших пристроїв (не тільки «Маяк-140», а й «Юпітер», «Олімп» тощо).

Апробація результатів здійснювалася шляхом демонстрації функціональних прототипів, проведення юзабіліті-тестування та аналізу відгуків користувачів, що дозволило підтвердити освітню та пізнавальну цінність створеної інтерактивної моделі.

Публікації.

1. Філіпас О. О. Використання тривимірних моделей і візуалізацій / О. О. Філіпас ; наук. кер. М. А. Мелешко // Мультимедійні технології в освіті та інших сферах діяльності : тези доп. наук.-практ. конф., м. Київ, [Дата проведення] 2024 р. – Київ : НАУ, 2024. – с. 274-276.

2. Веб-платформи та гейміфікація як інструменти соціальної мобілізації: від теорії до прикладних рішень. О. О. Філіпас ; наук. кер. О. А. Бобарчук // Мультимедійні технології в освіті та інших сферах діяльності : тези доп. наук.-практ. конф., м. Київ, [Дата проведення] 2025 р. – Київ. УДК 004.94:351.746.1(043.2).

3. Особливості реалізації аудіо-візуальної інтерактивності у веб-середовищі за допомогою Web Audio API та Three.js.

РОЗДІЛ 1

ТЕОРЕТИКО-МЕТОДОЛОГІЧНІ ОСНОВИ ІНТЕРАКТИВНИХ МУЗЕЙНИХ ЕКСПОЗИЦІЙ

1.1. Концептуальні засади мультимедіа

Поняття мультимедіа сформувалося не одномоментно – воно стало результатом тривалого розвитку технічних, художніх і освітніх підходів до подання інформації. Сучасне трактування мультимедіа, що поєднує текст, графіку, звук, відео й інтерактивні елементи, виникло на основі низки технологічних та культурних процесів другої половини ХХ століття. Хоча нині мультимедіа тісно пов'язане з цифровими платформами, його ідеї почали формуватися задовго до появи комп'ютерів, які ми знаємо сьогодні.

Перші прояви мультимедійності почали формуватися у мистецьких практиках 1960-х років, коли художники експериментували з поєднанням світла, звуку, фото- та відеоматеріалів у спільних перформансах [1]. Саме в цей час починає зміцнюватися й термінологічна основа сучасного поняття мультимедіа. Термін Мультимедіа (лат. *Multum + Medium*) – це одночасне використання різних засобів представлення інформації при розв'язанні завдань, зокрема тривимірної комп'ютерної графіки, звукового і відеоряду. Зараз можемо додати сюди ще й віртуальну або доповнену реальність [2]. Це визначення добре відображає природу ранніх мультимедійних експериментів: вони виходили за рамки одного медіуму й створювали синтетичне середовище, у якому різні засоби виразності взаємодіяли між собою.

Паралельно з мистецькими практиками розвивалися й освітні мультимедійні формати: діапроектори, аудіокасети, навчальні фільми. У цих матеріалах поєднували зображення, текст і звук, а основний акцент робився на тому, щоб навчальний матеріал був доступнішим і наочнішим.

Поява комп'ютерів стала тим моментом, коли мультимедіа перестало бути набором окремих пристроїв і перетворилося на цілісну технологічну систему. Якщо

до цього звук, світло чи відео існували поруч, але окремо, то комп'ютер уперше зібрав їх у єдиному середовищі. Він дав змогу зберігати й обробляти різні типи даних разом, а отже – створювати матеріали, де текст, графіка та звук взаємодіють між собою, а не просто “працюють поруч”.

Особливо помітним цей перехід став після появи Macintosh у 1984 році. Цей комп'ютер запропонував графічний інтерфейс, із яким мультимедіа стало більш “видимим” і доступним широкому колу користувачів. На ньому можна було створювати документи, працювати з зображеннями й аудіо в одному проєкті – для свого часу це було абсолютно новим форматом роботи з інформацією [3]. Фактично саме Мак став першим масовим інструментом, де мультимедіа стало практичним, а не концептуальним явищем.



Рис. 1.1. Macintosh, 1984

У 1990-х комп'ютери значно додали потужності, і це відкрило двері для обробки відео, анімації та перших форм тривимірної графіки. Поширення CD-ROM дало можливість створювати об'ємні цифрові довідники та інтерактивні навчальні програми, де зображення, звук і текст працювали разом. Це був важливий етап:

мультимедіа перестало бути лише демонстрацією й почало виконувати дидактичну та пояснювальну функції.

Важливу роль відіграло і програмне забезпечення. Такі середовища, як Macromedia Director [4], дозволили створювати інтерактивні презентації, а подальша поява Flash наприкінці 1990-х років суттєво змінила уявлення про те, яким може бути мультимедіа у вебсередовищі. До цього Інтернет складався переважно зі статичних сторінок, де можливості браузера обмежувалися текстом і простими зображеннями. Flash запропонував принципово новий підхід: об'єднання в одному середовищі векторної графіки, анімації, звуку, відео та інтерактивної логіки. Завдяки цьому веб уперше отримав інструмент, який дозволяв створювати динамічний, “живий” контент, що реагує на дії користувача. Фактично Flash перетворив браузер із засобу перегляду сторінок на платформу для інтерактивного досвіду.

У 2000-х саме Flash став основою ранньої веб-анімації та цифрового сторітелінгу [5]. У цьому середовищі з'являлися перші інтерактивні ролики, освітні симуляції, мініігри та анімовані інтерфейси. Flash був настільки поширений, що певний період більшість мультимедіа в Інтернеті створювали саме на ньому. Ключовим також було те, що Flash став основою для стрімінгового відео: перші великі відеоплатформи, включно з раннім YouTube, використовували Flash Player. Це визначило подальший розвиток вебвідео та зробило відеоконтент масовим явищем.

Важливо й те, що Flash сформував раннє уявлення про інтерактивність у цифрових середовищах. Він заклав принципи, які згодом перейшли у вебтехнології нового покоління: анімацію на сторінках, векторну графіку, реакції на взаємодію, мікроанімації та побудову інтерфейсів, що “звертаються” до користувача. Flash довів, що веб може бути не тільки текстовим, а й мультимедійним простором, де користувач активно впливає на те, що відбувається на екрані. І хоча згодом він поступився місцем відкритим технологіям (HTML5, WebGL, Canvas, JavaScript), саме він задав напрям, який визначив вигляд Інтернету на десятиліття вперед. Багато сучасних мультимедійних інструментів стали прямою відповіддю на ті можливості, які колись уперше продемонстрував Flash. У цьому сенсі Flash можна вважати ключовою

технологією, що підготувала перехід від статичного вебу до інтерактивного та мультимедійного.

Історично склалося так, що майже вся веб-анімація – від простих банерів до складних інтерактивних сцен – трималася на Adobe Flash. Це була епоха, коли Flash здавався незамінним: він давав мультимедіа там, де сам веб її ще не вмів. Але з часом його слабкі сторони стали надто очевидними: проблеми з безпекою, закритість екосистеми й повна несумісність із мобільними платформами поступово звели технологію нанівець. Фактично, веб був змушений вирости й навчитися працювати з мультимедіа нативно.

Цей перехід став ключовим моментом: відмова від Flash об'єднала зусилля розробників навколо відкритих стандартів, і саме це дало поштовх появі інструментів, якими ми користуємося сьогодні. HTML5 узяв на себе роботу з мультимедіа, а WebGL відкрив прямий доступ до графічного процесора через браузер – без плагінів, без костилів, прямо «на чистому вебі». Це був фактичний старт нової ери веб-3D.

На базі цих стандартів з'явилися бібліотеки, які зробили 3D доступним не лише інженерам, а й звичайним розробникам. Three.js – одна з них. Саме вона дозволяє побудувати цю роботу: інтерактивну 3D-модель магнітофона «Маяк-140», яка працює прямо в браузері, без додаткових встановлень. Те, що колись вимагало Flash та окремих плагінів, тепер реалізується легально, відкрито й підтримується всіма сучасними платформами.

З цього моменту мультимедіа стало не тільки тим, що показують, а й тим, що реагує на дії користувача. Інтерактивність закріпилася як невід'ємна частина мультимедійної екосистеми.

Після 2000-х років мультимедіа стрімко перемістилося у вебсередовище. Розвиток потокового відео, WebGL та бібліотеки Three.js дав змогу працювати з 3D прямо в браузері. Тепер для створення мультимедійної платформи не потрібно було спеціального обладнання – лише комп'ютер і доступ до Інтернету. Завдяки цьому мультимедіа стало мобільнішим і охопило значно ширшу аудиторію.

Ці зміни істотно вплинули на музейну сферу. Комп'ютери дали можливість створювати точні 3D-моделі технічних об'єктів, візуалізувати процеси, які

неможливо побачити у реальному експонаті, і робити це інтерактивно. Для техніки магнітного запису, де більшість механізмів працюють всередині корпусу, це може бути особливо важливо: саме мультимедіа дозволяє показати те, що приховане, і пояснити те, що складно відтворити наживо.

У підсумку комп'ютери не просто пришвидшили розвиток мультимедіа – вони змінили саму природу взаємодії з інформацією. Мультимедіа стало динамічним, інтерактивним і здатним пояснювати складні технологічні явища у доступній формі. Без комп'ютерів ця галузь навряд чи перетворилася б на те, чим вона є сьогодні: фундамент цифрових експозицій, інтерактивних музеїв і освітніх вебплатформ.

Мультимедіа перестає бути лише засобом передачі інформації – воно стає формою взаємодії. Відтепер користувач може не просто переглядати матеріал, а впливати на нього: обирати сценарії, змінювати параметри, взаємодіяти з об'єктами. Саме в цей час у музеях з'являються перші цифрові кіоски, інтерактивні панелі й мультимедійні гіди, що дозволяють відвідувачеві самостійно формувати траєкторію дослідження експонатів [6].

З 2010-х років акцент зміщується у бік інтерактивності й реалістичної візуалізації. Поява WebGL, а згодом бібліотеки Three.js, зробила можливим відтворення складних 3D-сцен у звичайному браузері без додаткових плагінів [7]. Паралельно розвиваються технології обробки аудіо в реальному часі, що дозволило поєднувати звук і графіку у динамічні візуалізації – спектрограми, еквалайзери, синхронізовані ефекти.

Завдяки цьому мультимедіа перестає бути ілюстрацією й стає інструментом пояснення. Воно дає можливість буквально «побачити» фізичні або технічні явища, які неможливо відтворити в реальному експонаті.

1.1.1. Сутність, компоненти та дидактичні функції мультимедіа

Структурно мультимедіа складається з кількох взаємопов'язаних компонентів. Текст забезпечує передачу фактологічної інформації; графічні матеріали формують наочність; аудіо підсилює емоційний і когнітивний ефект; відео дозволяє показати

динаміку; а інтерактивність робить користувача активним учасником процесу. Сумісна робота цих компонентів створює «багатошарову» модель подання інформації, де кожен елемент доповнює інші та виконує власну функцію [8].

Таблиця 1.1

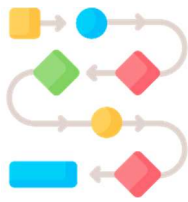
Компоненти мультимедійної системи

Компонент	Канал сприйняття	Призначення
Текст	Візуальний, мовний	Передача точних даних, визначень, інструкцій.
Графіка (Зображення)	Візуальний, просторовий	Передача статичної інформації, схем, діаграм, ілюстрацій.
Аудіо (Звук)	Слуховий	Музичний супровід, голосові пояснення (закадровий текст), імітація звуків процесів.
Відео та Анімація	Візуальний, часовий	Демонстрація динамічних подій, процесів, що відбуваються в часі (наприклад, рух стрічки магнітофону).

Дидактичні можливості мультимедіа тісно пов'язані з особливостями людського сприйняття. Дослідження в галузі когнітивної психології (Mayer, Moreno) [9] показують, що комбіновані канали подання інформації покращують розуміння, сприяють формуванню глибших зв'язків між фрагментами знань і зменшують когнітивне навантаження, якщо елементи підібрані коректно. Завдяки цьому мультимедіа набули важливого значення у освітньому процесі, музейній педагогіці та популяризації науково-технічних знань.

У контексті технічних експозицій мультимедіа виконують дві ключові функції. Перша – пояснення складних процесів, які важко або неможливо продемонструвати в реальному середовищі. Мультимедійні елементи дозволяють розкласти технічне явище на зрозумілі візуальні частини: схема → механізм → анімація → інтерактив.

Це особливо актуально у випадку магнітного запису, де значна частина процесу прихована всередині пристрою й недоступна для прямого спостереження. Завдяки мультимедіа користувач може побачити те, що у фізичному експонаті залишається невидимим.



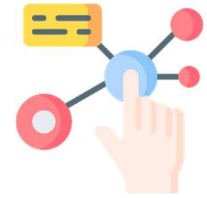
Схема



Механізм



Анімація



Інтерактив

Рис. 1.2. Мультимедійні рівні пояснення складного технічного процесу

Друга важлива функція – підсилення залученості аудиторії. Технічні об’єкти зазвичай сприймаються складними, і між відвідувачем та експонатом виникає певний бар’єр «технічності». Візуалізація допомагає цей бар’єр подолати: вона робить механізми ближчими, зрозумілішими й більш “людяними”. Інтерактивні моделі дозволяють користувачеві досліджувати об’єкт у власному темпі, переглядати його з різних ракурсів, самостійно обирати деталі, на які варто звернути увагу. Така свобода взаємодії позитивно впливає на мотивацію та рівень зацікавленості.

У підсумку мультимедіа виступає не лише технологією, а й методом сучасної комунікації. Її використання допомагає музеям поєднати інформативність із емоційністю, зробити технічні експонати доступнішими для різних груп відвідувачів і створює умови для глибшого розуміння складних технічних процесів. Саме тому мультимедійні рішення стали ключовим компонентом сучасних цифрових і гібридних музейних експозицій.

1.1.2. Психолого-педагогічні основи мультимедійного навчання

Психолого-педагогічні основи мультимедійного навчання спираються на уявлення про те, як людина сприймає й опрацьовує інформацію, коли вона надходить одночасно з різних сенсорних каналів. У центрі цих підходів – дві впливові концепції: Когнітивна теорія мультимедійного навчання Річарда Мейєра та Теорія когнітивного навантаження Джона Свеллера. Разом вони формують базу для розуміння того, як слід проєктувати мультимедійні матеріали, щоб користувач дійсно засвоював зміст, а не перевантажувався зовнішніми деталями.

Когнітивна теорія мультимедійного навчання (R. Mayer) описує мультимедійне навчання через три ключові припущення. Перше – подвійне кодування: людина обробляє вербальну інформацію (слова, аудіопояснення) через один канал, а візуальну (зображення, графіка, анімація) – через інший. Коли ці канали працюють разом, розуміння поглиблюється. Друге припущення – обмежена ємність робочої пам'яті: і візуальний, і слуховий канали мають свою межу, тому перенасичені матеріали лише шкодять навчанню. Третє – активне опрацювання: учень дійсно навчається лише тоді, коли сам обирає, структурує та інтегрує інформацію у власну ментальну модель [10].

На основі цих припущень сформувалися відомі принципи мультимедійного навчання Мейєра. Принцип мультимедіа підкреслює, що поєднання слів і зображень ефективніше, ніж використання лише тексту – саме тому аудіопояснення у 3D-моделі «Маяк-140» працюють значно краще, ніж довгі текстові блоки. Принцип когерентності вимагає прибирати все зайве – урізані підказки, стриманий інтерфейс і мінімум декоративних елементів допомагають користувачу зосередитися на процесі. Принцип модальності пояснює, що якщо модель уже зайняла візуальний канал, то текст слід замінювати озвученням, щоб не створювати конкуренції між каналами. Із цим пов'язаний і принцип просторової суміжності: коментарі та візуальні елементи повинні бути розташовані поруч, наприклад підказка «Play» має бути біля самої кнопки, а не десь осторонь. Нарешті, принцип сегментування пропонує подавати матеріал порціями, дозволяючи користувачу контролювати темп. У випадку магнітофону це може бути покрокове пояснення: схема, механізм, анімація, інтерактив.

Теорія когнітивного навантаження Свеллера (J. Sweller) деталізує, чому важливо грамотно організовувати мультимедійний контент. Робоча пам'ять має обмежений обсяг, тому навчальний матеріал може викликати три типи когнітивного навантаження. Внутрішнє навантаження визначається складністю самого змісту – наприклад, фізичні основи магнітного запису є складними незалежно від того, як вони подані. Це навантаження не можна прибрати, але можна зробити контрольованим, розбивши матеріал на логічні частини. Стороннє (extraneous) виникає через невдалий

дизайн – перевантажені інтерфейси, незрозумілу навігацію, повтори чи «візуальний шум». Його необхідно зменшувати через якісний UX/UI. Натомість продуктивне (germane) навантаження спрямоване на формування нових знань і ментальних моделей; його, навпаки, варто підтримувати – за допомогою інтерактивності, експериментування та можливості керувати демонстрацією [11].

Погляди Мейера та Свеллера добре узгоджуються з конструктивістськими підходами, які розвивав Девід Джонассен. У конструктивізмі навчання розглядається як активний процес: учні не просто отримують інформацію, а самі «будують» знання через дію, вибір і взаємодію [12]. Інтерактивна 3D-модель магнітофону якраз і створює таке середовище. Користувач може натискати кнопки, обертати котушки, регулювати еквалайзер і спостерігати, як змінюється робота пристрою. Фактично він взаємодіє з контентом так само, як із фізичним об'єктом, але має доступ до додаткових рівнів – наприклад, «рентгенівського» режиму чи покрокового пояснення процесів.

Усе це робить мультимедійну платформу не просто інтерактивним додатком, а інструментом, який поєднує когнітивні теорії з конструктивістською логікою навчання. Завдяки цьому користувач не лише дивиться на модель, а активно будує розуміння принципів її роботи, що і є кінцевою метою мультимедійного навчання.

1.1.3. Візуалізація як ключовий компонент мультимедіа

Візуалізація є одним із найважливіших елементів мультимедійного середовища, адже саме через зоровий канал людина отримує найбільше інформації. Як зазначає Едвард Тафті, *«добра візуалізація – це спосіб думати через графіку»* [13]. Вона не лише прикрашає мультимедійний контент, а й формує його структуру, допомагає побачити зв'язки та зрозуміти складні явища. У технічних експозиціях це особливо важливо, оскільки багато процесів або надто швидкі, або приховані в корпусі пристрою, і візуалізація фактично стає єдиним способом пояснити їхню природу.

Однією з ключових функцій візуалізації є подолання когнітивного бар'єра. Шнотц і Лоу підкреслюють, що динамічна графіка *«допомагає перетворювати*

абстрактні концепції на осмислені ментальні моделі» [14]. У випадку електромагнетизму, частотного спектра чи внутрішньої роботи магнітофона візуалізація «перекладає» складні фізичні процеси мовою образів, які легше сприймати. Завдяки цьому зменшується внутрішнє когнітивне навантаження, і користувач краще концентрується на суті явища.

Не менш важливою є здатність візуалізації стимулювати просторове мислення. Колін Веар зазначає: *«просторове уявлення є центральним механізмом людського мислення»* [15]. Це добре проявляється в 3D-моделях, де користувач може бачити, як компоненти технічного пристрою взаєморозташовані в просторі. На прикладі магнітофона «Маяк-140» тривимірна модель дозволяє сформувати чітке уявлення про взаємодію головки, стрічки, котушок та механіки, що в реальному експонаті майже недоступно через конструктивні обмеження.

Візуалізація також виконує функцію виявлення закономірностей. Як показують дослідження Лоу [16], анімації та візуальні графіки допомагають швидко помітити тенденції або аномалії, які губляться в текстових описах чи числових таблицях. Спектрограми, коливання сигналу, зміна частот у реальному часі – усе це стає зрозумілим саме завдяки візуальному поданню.

Ще один ефект – підсилення запам'ятовування. Тафті пише, що люди *«запам'ятовують образи значно краще, ніж абстрактні символи»* (Tufte, 1990), а сучасні когнітивні дослідження підтверджують, що візуальна інформація створює потужні асоціативні зв'язки, які тримаються довше.

У мультимедіа візуалізація реалізується в різних формах: статичні схеми, динамічні анімації, відео, інтерактивна графіка та 3D-моделі. Кожна з них має свою функцію. Статичні зображення добре підходять для відображення структури; анімація дає змогу «розгорнути» процес у часі; відео демонструє реальні явища; інтерактивна графіка залучає користувача до дослідження; а 3D-візуалізація створює імерсивний простір, де можна взаємодіяти з моделлю так, ніби це реальний об'єкт.

Особливо ефективною є 3D-візуалізація технічних процесів. Вона забезпечує високий рівень реалізму, дозволяє показувати внутрішню будову пристрою («рентгенівські» режими), і робить можливим моделювання процесів, які неможливо

побачити наживо. Наприклад, у моделі «Маяк-140» можна продемонструвати поведінку магнітної стрічки, роботу головки, зміни сигналу та будову внутрішніх вузлів – і все це в інтерактивному форматі. У результаті користувач не просто спостерігає за моделлю, а активно досліджує її, що значно підсилює формування ментальних моделей.

Таким чином, візуалізація в мультимедіа працює на кількох рівнях: пояснює, структурує, залучає та робить знання стійкішими. У музейних експозиціях це перетворює складні технічні об'єкти на зрозумілі й доступні, а користувача – з пасивного спостерігача на активного дослідника. Сучасні наукові дослідження, зокрема матеріали XV Міжнародної науково-практичної конференції, організованої ДУ «КАІ» (НАУ) та публікації, присвячені імерсивним технологіям в освіті (2024-2025 рр.), підтверджують високу актуальність використання 3D-моделювання та інтерактивних засобів у сфері культурної спадщини та освіти [45].

1.2.Інтерактивність, проблемне навчання та музейна педагогіка

Сучасні музейні експозиції дедалі частіше будуються на ідеї активної участі відвідувача. Зокрема Дональд Джонассен [12] виводить принципи «проблемного навчання», де люди навчаються найефективніше тоді, коли опиняються в ситуації, яка потребує вирішення реальної проблеми. Замість отримання готових фактів, учень стикається з певною невизначеністю, аналізує умови, порівнює можливі варіанти та самостійно будує рішення. Така взаємодія активує глибші когнітивні процеси, оскільки людина не просто сприймає інформацію, а намагається зрозуміти, чому щось працює або не працює. За Джонассеном, навчання є ефективним тоді, коли учень сам бере участь у процесі: взаємодіє з матеріалом, приймає рішення, впливає на перебіг подій і бачить наслідки своїх дій. Це перетворює його з пасивного спостерігача на активного учасника, який конструює знання, а не просто споживає їх. Саме тому інтерактивність відіграє ключову роль у сучасних цифрових середовищах.

Цей принцип напряду стосується інтерактивних музейних експозицій. Коли відвідувач натискає кнопку, змінює параметри чи запускає демонстрацію і одразу

бачить результат, він діє в логіці проблемного навчання, описаній Джонассеном. Інтерактивний експонат створює проблемну ситуацію, стимулює цікавість і залучає до експерименту, що приводить до більш глибокого розуміння та кращого запам'ятовування матеріалу.

Цифрові технології дозволили відійти від моделі пасивного огляду й сформували новий формат взаємодії, у якому користувач виступає не спостерігачем, а дослідником. Інтерактивність, проблемне навчання та музейна педагогіка працюють разом, створюючи умови для осмисленого пізнання та особистісного залучення.

1.2.1. Інтерактивність як спосіб залучення та формування проблемного поля

Інтерактивність у мультимедійних експозиціях – це не просто технічна функція, а засіб формування активного пізнання. Користувач отримує можливість взаємодіяти з цифровим експонатом, змінювати параметри, запускати демонстрації й одразу бачити результат своїх дій. Така модель взаємодії природно створює проблемне поле, у якому відвідувач стикається з питаннями, що потребують аналізу та пояснення. Наприклад, регулюючи еквалайзер або спостерігаючи за роботою стрічкопротяжного механізму у 3D-моделі «Маяк-140», користувач сам помічає зміни, формулює гіпотези та перевіряє їх, що відповідає логіці проблемного навчання.

Інтерактивність забезпечує й індивідуалізацію: кожен відвідувач може йти у власному темпі, повторювати дії, досліджувати альтернативні сценарії, не боячись «помилитися». Це створює ефект лабораторної свободи та відчуття, що експозиція працює «для нього».

З позиції психології сприйняття інтерактивність активує ефект володіння (Endowment Effect): користувач сильніше цінує досвід, коли відчуває, що він «керує» експонатом або досягає певного результату завдяки власним діям [17]. Коли відвідувач самостійно знаходить правильну частоту або запускає механізм, його мотивація та зацікавленість зростають. Важливу роль відіграє й теорія прогалин (Information Gap Theory): неповна або фрагментована інформація створює природну цікавість [18]. Якщо модель показує «проблему» (наприклад, перекручення стрічки

чи зміну спектра), але не дає готового пояснення, користувач прагне заповнити цю прогалину й починає експериментувати з моделлю.

1.2.2. Музейна педагогіка в епоху цифрових технологій та концепція «Digital Heritage»

Цифрові технології радикально трансформували музейну педагогіку. Музей перестав бути лише сховищем артефактів і став гнучким освітнім середовищем, де контент подається через мультимедіа, інтерактивність і цифрові наративи. Концепція Digital Heritage охоплює все – від оцифрованих 3D-моделей до віртуальних реконструкцій і онлайн-архівів, що дозволяють зберігати й поширювати культурну та технічну спадщину у глобальному масштабі [19].

Однак ключовим завданням цифрової спадщини є не тільки передати форму об'єкта, а й забезпечити аутентичність цифрового досвіду. Для технічних експонатів це подвійний виклик: потрібно зберегти як візуальну, так і функціональну точність. У випадку магнітофона «Маяк-140» це означає, що користувач має бачити не лише реалістичну 3D-модель, а й взаємодіяти з нею так, як із реальним пристроєм. Саме тому у веб-платформі використовується Web Audio API для коректної обробки сигналу та фізично правдоподібна анімація руху стрічки. Така функціональна аутентичність формує довіру користувача й підвищує освітню ефективність.

Цифрові музеї також все частіше використовують механізми персоналізації: цифрові аватари, інтегровані гіди, чат-боти або інтерактивні підказки, які адаптують зміст і рівень складності під потреби різних груп аудиторії. На основі аналізу поведінки користувача штучний інтелект може рекомендувати додаткові пояснення, міні-ігри, альтернативні маршрути чи поглиблену інформацію. Це створює індивідуальний «освітній шлях» для кожного відвідувача.

Гейміфікація доповнює цей підхід. Квести, збір досягнень, розв'язування технічних головоломок і навіть віртуальні нагороди перетворюють відвідування цифрового музею на захопливу гру. Гейміфіковані експозиції підтримують увагу довше й роблять навчальний процес привабливим для молодшої аудиторії.

Сучасні Digital Heritage-проекти також розширюють можливості співтворчості. В окремих музеях користувачі можуть додавати власні коментарі, анотації або гіпотези до 3D-моделей – так працюють, наприклад, віртуальні конструкторські лабораторії British Museum [20]. Це формує децентралізоване середовище знань і сприяє активному осмисленню експонатів.

1.2.3. Когнітивний аспект та досвід користувача (UX) у музейному середовищі

У цифровому музеї досвід користувача стає не менш важливим, ніж сам контент. UX визначає, як саме користувач опрацьовує інформацію, чи розуміє він логіку експозиції й наскільки глибоко залучений у вивчення матеріалу. Якісний UX зменшує стороннє когнітивне навантаження й підтримує продуктивне: структура має бути простою, інтерфейс – передбачуваним, а навігація – логічною [21].

Особливе місце посідає принцип когнітивної доступності. У музеї можуть бути відвідувачі з різним досвідом, рівнем технічної підготовки чи навіть різними здібностями. Інтерфейс повинен адаптуватися: для когось – спрощена 2D-схема, для інших – детальна 3D-модель із технічними параметрами. Це розширює охоплення аудиторії та робить музейний контент інклюзивним.

Інклюзивність – загалом одна з ключових вимог сучасного UX. У цифровому середовищі експозиція стає доступною незалежно від географії та часу [22]. Адаптивні інтерфейси пропонують альтернативні режими візуалізації, озвучення, субтитри та навіть тактильні 3D-моделі для людей із порушеннями зору. Digital Heritage-проекти дедалі частіше враховують ці аспекти при створенні інтерактивних платформ.

Досвід користувача також підсилюється завдяки AR і VR. Доповнена реальність дозволяє «перенести» модель у простір відвідувача, а VR – занурює його у відтворений мікросвіт технічного пристрою або історичний контекст. Це відкриває можливості демонструвати експонати, які фізично недоступні або занадто делікатні для огляду.

Ще один аспект – мультиканальність і адаптивний UX. Сучасні системи дозволяють користувачу почати знайомство з експонатом удома на ноутбучі, продовжити його у смартфоні в дорозі чи поглибити знання вже в музеї за допомогою сенсорної панелі. Усі дані синхронізуються, створюючи безшовний досвід, який підлаштовується під стиль життя людини.

Серед яскравих прикладів сучасного UX у музейній сфері можна назвати:

- VR-тур Smithsonian National Museum of Natural History [23];
- платформу British Museum для колективного анотованого перегляду 3D-об'єктів [24];
- сенсорні панелі та адаптивні інтерфейси MIM у Брюсселі, які підлаштовують складність пояснень до віку відвідувача [25].

UX у цифрових музеях – це не лише про «зручність», а про створення освітньо-когнітивного середовища, яке дає користувачеві інструменти для дослідження, експериментування, формування власних висновків і збереження інтересу до складних технічних тем.

1.3. Аналіз складних технічних процесів (на прикладі магнітного запису)

Технічні процеси – особливо ті, що стосуються звукозапису – завжди мали репутацію «невидимих». Рух стрічки, зміна магнітного поля, трансформація сигналу – усе це відбувається всередині пристрою й лишається поза сприйняттям відвідувача. Через це навіть простий магнітофон виглядає як «чорна скринька»: працює – і добре, а як саме – зрозуміти непросто. Саме тому для створення якісної інтерактивної моделі необхідно не просто показати прилад у 3D, а розібратися в його фізиці, перетворити складні процеси на візуально зрозумілу історію й адаптувати її під користувача різного рівня підготовки.

Розділ 1.3 пояснює три ключові речі: як працює магнітний запис і чому це важливо для моделювання, чому 3D-візуалізація – найбільш ефективний спосіб «побачити невидиме», і які вимоги стоять перед UX-механікою технічних експонатів,

що містять багато абстрактних даних (частоти, спектри, електричні сигнали, фізичні явища).

1.3.1. Фізичні та технічні принципи магнітного запису та відтворення (як основа для моделювання)

Основою магнітного запису є здатність феромагнітних частинок на стрічці змінювати орієнтацію під впливом магнітного поля. Коли на записувальну головку подається електричний аудіосигнал, у ній виникає змінне магнітне поле, яке «впорядковує» частинки на стрічці – так формується магнітний еквівалент звуку. При відтворенні процес працює навпаки: вже намагнічена стрічка змінює магнітний потік у відтворювальній головці, і цей потік перетворюється назад на електричний сигнал.

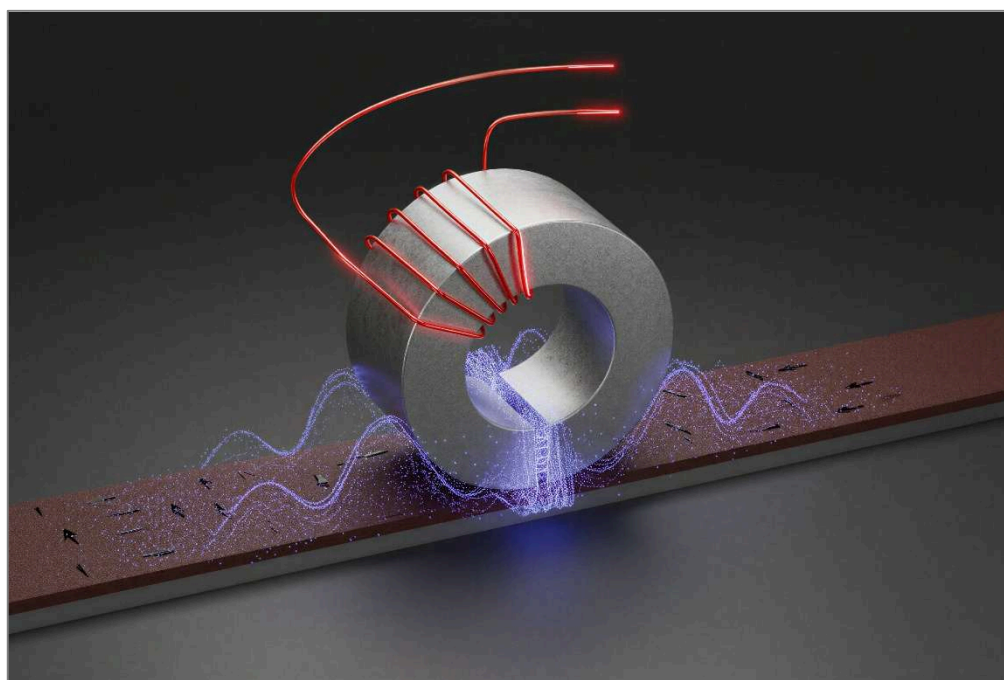


Рис. 1.3. Принцип магнітного запису

Однак коли ми намагаємося перенести роботу магнітофона в інтерактивну 3D-модель, одразу впираємося в одну просту, але жорстку реальність: те, що відбувається зі стрічкою й доменами, занадто дрібне і занадто швидке, щоб це можна було показати в реальному часі. Перемагнічування відбувається на мікроскопічному рівні, а частота

коливань сигналу сягає десятків тисяч разів на секунду. Якщо спробувати «показати» це візуально, довелося б уповільнити процес у тисячі разів – і ми б отримали красиву, але відірвану від реальної музики анімацію. Користувач би бачив одне, чув інше – і весь сенс мультимедійної синхронізації зник би.

Тому для реалістичного відображення роботи «Маяка-140» обрано більш чесний і водночас автентичний шлях – використати вакуумно-люмінесцентний індикатор рівня (VFD). Це не вигадана графіка, а реальний елемент панелі магнітофона, який історично відповідає за контроль рівня сигналу. Тобто ми не просто спрощуємо процес – ми переносимо його у форму, якою користувалися ще в 1980-х.



Рис. 1.4. Маяк-140 з вакуумно-люмінесцентним індикатором рівня (VFD)

Чому саме VFD працює ідеально? Він наочно показує фізику сигналу. Стовпчики індикатора «малюють» амплітуду звуку в децибелах. І тут відразу стають очевидними базові речі:

- 0 дБ – це номінальна гучність;
- +3...+5 дБ – це перевантаження і спотворення;
- занадто низький рівень означає шум та втрату деталей.

І найважливіше – користувач бачить ці зміни буквально в момент, коли вони відбуваються.

Він миттєво реагує на звук. На відміну від доменів у стрічці, які фізично не можна «показати» в реальному часі, VFD реагує одразу. Натиснув «Play», пішов гучний акорд – індикатор вибухає вгору. Це та сама мультимедійна когерентність, про яку пише Маєр: звук і візуальний канал працюють разом, без затримок і фальші.

Він виглядає круто (і це теж важливо). Фірмове зелене світіння на чорному тлі – це чиста естетика hi-fi 80-х. Сьогодні вона зчитується як суміш ретро-футуризму й легкого кіберпанку, тому індикатор не просто функціональний, а й дуже виразний візуально.

Є в роботі VFD-індикатора ще один феномен, який часто недооцінюють: він працює на тій самій психології, що й звичайний камін. Коли ми дивимося на полум'я, увага ніби сама «приклеюється» до його ритму й мікрорухів – нічого особливо не відбувається, але відвести погляд важко. Із VFD-сегментами відбувається майже те саме. Вони мерехтять, дихають ритмом музики, підкидають короткі яскраві спалахи на піках сигналу – і цей візуальний «живий вогонь» тримає увагу користувача без жодних додаткових стимулів. Це створює майже медитативний ефект: людина не просто дивиться на прилад, вона «співвідчуває» музику через рух світла. І саме завдяки цьому індикатор стає не лише технічним елементом, а й емоційним центром взаємодії з моделлю.

1.3.2. Обґрунтування використання 3D-візуалізації для демонстрації невидимих процесів

Важливо пам'ятати, що вся магнітна «кухня» – це робота електричних сигналів і мікроскопічних магнітних доменів, яких людське око просто не здатне побачити. Ми не можемо спостерігати, як змінюється напруга в проводах чи як орієнтуються частинки на стрічці. Тому 3D-візуалізація панелі керування та індикаторів стає найкращим способом перетворити абстрактну фізику на зрозумілий візуальний досвід – буквально зробити невидиме видимим.

У цифровій моделі «Маяка-140» особливу роль відіграє аудіо-візуальна синергія, тобто взаємодія звуку та VFD-індикації, коли обоє каналів працюють разом.

Вакуумно-люмінесцентний індикатор (VFD) виступає візуальним дублером аудіосигналу і реалізує принцип подвійного кодування Маєра: користувач одночасно отримує інформацію слухом і зором. Сценарій простий: користувач піднімає низькі частоти на еквайзері → звук одразу змінюється → індикатор стрибає вище. Така трійка «дія → звук → світло» створює відчутну причинно-наслідкову логіку, а складні терміни на кшталт «амплітуда» та «динамічний діапазон» раптом стають інтуїтивно зрозумілими.

У віртуальній моделі є ще одна перевага – можливість безпечної симуляції граничних станів. У реальній аналоговій техніці перевантаження (clipping) – це ризик спотворити запис або навіть пошкодити обладнання. У цифровому середовищі все навпаки: користувач може навмисно «загнати» сигнал у червону зону, подивитися, як індикатор залипає на +3...+5 дБ, і почути, як звук починає «хрипіти». Це дозволяє навчитися тому, що в реальності краще взагалі не пробувати. Виходить чесний, живий і абсолютно безпечний досвід роботи з «небезпечними» режимами.

І, нарешті, важливий аспект – імерсивність та естетичне залучення. Тривимірний VFD-індикатор у браузері веде себе майже як справжній: має глибину, характерне зелене світіння й легкий «glow», змінює вигляд залежно від кута огляду. Завдяки цьому виникає ефект присутності – ніби ти стоїш перед справжнім магнітофоном. І тут починається психологія: людина дивиться на індикатор так само, як на камін – світіння, що “дихає” разом із музикою, утримує увагу природним чином. Це не суха схема, а жива емоційна точка експозиції, що поєднує звук і світло в єдиний інтерактивний об’єкт.

У підсумку 3D-візуалізація у цій роботі – не декоративний шар, а повноцінний когнітивний інструмент. Вона пояснює складну фізику, підсилює розуміння через синхронність слуху й зору, дозволяє безпечно експериментувати з крайніми режимами та створює залучення, яке важко досягти звичайними плоскими інтерфейсами. Саме так музейна експозиція перетворюється з технічної інформації на захопливий досвід.

1.3.3. Вимоги до юзабіліті та доступності інтерактивних технічних експонатів

Навіть якщо ми переносимо інтерфейс магнітофона «Маяк-140» у цифрове середовище буквально один в один – зі всіма кнопками, повзунками та перемикачами, – це не означає, що якусь роботу UX можна пропустити. Фізичний прилад і браузер – це два різні світи, і те, що інтуїтивно зрозуміле руками, не завжди так само зрозуміле у вигляді 3D-моделі. Тому навіть при збереженні автентики нам потрібні базові принципи UX, які допоможуть користувачеві орієнтуватися у складній ретро-панелі та взаємодіяти з нею без плутанини.

Наприклад, важливо зберегти скевоморфічний характер елементів, щоб повзунки виглядали як реальні повзунки, а кнопки – як об'єкти, які можна натиснути. Ми не змінюємо їхню форму, не переробляємо інтерфейс – ми просто підсилюємо фізичну логіку елементів, яку людина очікує побачити у 3D. Це дозволяє користувачу спиратися на вже знайомий досвід: у житті ручка гучності крутиться, у моделі вона теж повинна крутитись, а не бути просто текстурою.

Те саме стосується питання орієнтації в просторі. У реальному житті користувач стоїть перед панеллю під певним кутом, на фіксованій дистанції. У браузері ж камера може «вилетіти», перевернутись або відійти надто далеко. Щоб цього не сталося, камера має бути обмежена так, щоб користувач завжди сприймав пристрій так, як він сприймав би його в реальному житті. Це не зміна інтерфейсу, а збереження його читабельності.

Інтерактивні елементи також потребують мінімального позначення: у реальному приладі ти відчуваєш кнопку пальцями, у браузері – цього немає. Тому при наведенні варто додати легке світлове підкреслення або зміну курсору, щоб користувач розумів: «ось цим елементом я можу керувати». Це не модернізація «Маяка», а адаптація фізичного досвіду до цифрового простору.

Ключова частина UX – миттєвий зворотний зв'язок, і тут основну роль відіграє VFD-індикатор. У реальному пристрої він і був тим каналом, що показує стан сигналу «тут і зараз». У моделі він виконує те саме завдання: підтверджує, що кнопка

спрацювала, звук пішов, рівень змінився. Це цілком автентична поведінка, просто перенесена у 3D.

Ще один важливий момент – технічна доступність. Ми не змінюємо інтерфейс, але мусимо забезпечити, щоб він працював на різних пристроях: комп'ютерах, смартфонах, планшетах. Фізичний магнітофон універсальний сам по собі, а ось цифрова модель має гнучко підлаштовуватися під залізо користувача. Це питання оптимізації, а не редизайну.

І, нарешті, корисно дати мінімальний онбординг. Людина може й не знати, що означає той чи інший повзунок еквайзера або чому кнопка «Рес» не працює без «Play». Короткі підказки не змінюють інтерфейс, вони лише пояснюють, як ним користуватися – так, як це зробив би консультант у музеї.

Тож навіть якщо інтерфейс переноситься 1:1, завдання UX – це не перепрацювати «Маяк», а забезпечити максимально автентичний, але зрозумілий досвід у цифровому середовищі, де немає тактильного зворотного зв'язку, фізичних обмежень камери та інтуїтивного «відчуття» реального приладу.

1.4.Методології створення інтерактивних навчальних експонатів

Створення якісного цифрового музейного експонату – це не лише програмна реалізація, а передусім комплексне проектування взаємодії, яке вимагає чітких методологій. У розробці веб-платформи для магнітофона «Маяк-140» застосовано три ключові підходи, базовим з яких є ітеративне проектування (Iterative Design) [26]. Оскільки для складних інтерактивних 3D-продуктів лінійні методи розробки є малоефективними, процес доцільно розбивати на короткі цикли – спринти. Робота починається з етапу Greyboxing, де створюється чорновий прототип із геометричних примітивів для тестування механіки та камери. Далі слідує етап High-fidelity, що передбачає додавання детальної геометрії, PBR-матеріалів та налаштування освітлення для перевірки візуальної автентичності. Фінальним кроком є функціональна інтеграція, коли візуальна модель підключається до логіки JavaScript

та Web Audio API. Такий підхід дозволяє виявляти помилки юзабіліті на ранніх стадіях і суттєво економити ресурси.

Щоб утримати увагу користувача та уникнути когнітивного перевантаження, наступним кроком впроваджується сценарний підхід та гейміфікація [27]. Залишити відвідувача сам на сам зі складною панеллю керування – ризиковано, тому система використовує продумані User Journeys. Взаємодія починається зі сценарію «Онбординг», де інтерфейс підсвічує ключові елементи управління, знижуючи поріг входу. Після цього активується режим «Вільне дослідження» (Sandbox), що доповнюється елементами гейміфікації. Наприклад, налаштування рівня запису перетворюється на інтерактивний челендж: користувач має збалансувати звук, орієнтуючись на VFD-індикатор, щоб уникнути «червоної зони» перевантаження. Це трансформує суху технічну операцію на зрозумілу ігрову механіку.

Не менш важливою умовою сучасного веб-проєкту є забезпечення вебдоступності (Web Accessibility) згідно зі стандартами WCAG 2.1 [28]. Музейний експонат має бути інклюзивним, тому для 3D-середовища розробляються механізми альтернативної взаємодії. Це включає навігацію з клавіатури (перемикання між інтерактивними зонами клавішею Tab), використання семантичної розмітки для коректної роботи скрінрідерів (озвучення функцій кнопок), а також забезпечення достатньої контрастності елементів VFD-індикатора для користувачів із порушеннями зору. Використання цих методологій дозволяє створити стабільний, зрозумілий та доступний освітній продукт.

Висновки до розділу 1

У першому розділі ми проаналізували теоретичну базу та методологію створення цифрових музейних експонатів.

По-перше, технології готові. Еволюція вебу від Flash до нативного WebGL дозволяє створювати складну 3D-графіку прямо в браузері. Це відкриває шлях до створення глобально доступних віртуальних музеїв без необхідності встановлювати додатковий софт.

По-друге, візуалізація має бути макроскопічною. Замість того щоб показувати невидимі процеси на мікрорівні (намагнічування доменів), обрано шлях аудіовізуальної синергії через VFD-індикатор. Це дозволяє реалізувати принцип подвійного кодування: користувач чує зміни в звуці й одночасно бачить їх на екрані. Це робить абстрактну фізику зрозумілою.

По-третє, інтерактивність – це ключ. Відходимо від пасивного споглядання до активного дослідження. Можливість безпечно «перевантажити» сигнал або розібратися з механікою через 3D-модель дає досвід, недоступний у звичайному музеї.

І наостанок, визначили правила розробки. Використання ітеративного підходу, сценаріїв та стандартів доступності – це фундамент, на якому будуватиметься наша платформа.

РОЗДІЛ 2

АНАЛІЗ ТА ПРОЄКТУВАННЯ ТЕХНОЛОГІЧНИХ ЗАСОБІВ ВЕБ-ПЛАТФОРМИ

У попередньому розділі було визначено, що для ефективної цифрової музейної експозиції критично важливими є інтерактивність та реалістична візуалізація процесів, які зазвичай залишаються невидимими всередині технічного об'єкта. Наступним кроком є перехід від теоретичних вимог до їхньої практичної реалізації. Саме цьому присвячено другий розділ – вибору й обґрунтуванню технологічного стеку, який дозволить створити веб-платформу з максимальною продуктивністю та доступністю для користувача.

У межах цього розділу буде проведено аналіз сучасних інструментів для 3D-моделювання та веб-візуалізації, щоб знайти оптимальний баланс між якістю зображення та швидкістю завантаження. Окрему увагу приділено технологіям роботи зі звуком у реальному часі, адже саме вони забезпечують «живу» реакцію моделі магнітофона на дії користувача. Від правильного вибору цих інструментів залежить не лише зовнішній вигляд експоната, а й його стабільність на різних пристроях і браузерах.

Крім того, у цьому розділі буде спроектовано архітектуру вебдодатка та визначено методику створення 3D-моделі «Маяка-140». Ми послідовно пройдемо шлях від вибору формату файлів до побудови сценаріїв взаємодії, заклавши технічний фундамент для практичної реалізації, яка детально описується у третьому розділі.

2.1. Технологічний стек для створення та відтворення 3D-моделі

Створення технічно точного музейного експоната вимагає вибору такого програмного забезпечення, яке дозволяє не лише детально опрацьовувати механічні вузли, а й ефективно оптимізувати геометрію для веб-середовища. Для реалізації

проекту було розглянуто три найбільш поширені полігональні редактори (Blender, Autodesk Maya, Cinema 4D), а також оцінено доцільність використання САД-систем.

2.1.1. Порівняння інструментів 3D-моделювання та обґрунтування вибору Blender

Autodesk Maya є індустріальним стандартом у кінематографі та анімації. Вона пропонує потужні засоби для рігінгу та процедурної анімації. Проте в контексті даного проекту виявлено низку суттєвих недоліків [29]:

- Надлишкова функціональність: Фокус Maya на кінематографічних пайплайнах є надмірним для створення статичного або обмежено інтерактивного веб-експоната.
- Складність експорту: Підготовка моделі до WebGL (формат glTF) часто потребує сторонніх плагінів або складних скриптів.
- Економічна недоцільність: Висока вартість ліцензії не виправдовує себе в рамках бюджетних обмежень проекту.

Cinema 4D Цей редактор є стандартом у motion-дизайні та рекламі, вирізняючись низьким порогом входу та зручністю створення анімованої графіки. Однак для технічного моделювання його можливості обмежені [30]:

- Специфіка моделювання: Інструментарій Cinema 4D менш орієнтований на точне інженерне («hard surface») моделювання порівняно з конкурентами.
- Проблеми з передачею матеріалів: Експорт у glTF часто потребує додаткових налаштувань для коректного відображення PBR-матеріалів у веб-браузері.

SolidWorks, Fusion 360. Хоча інженерні САД-системи ідеальні для креслень, вони працюють з NURBS-поверхнями. При конвертації у полігональну сітку (для вебу) це часто призводить до неоптимальної топології, артефактів затінення та надмірної ваги файлу, що є критичним для швидкодії веб-додатку.

За результатами аналізу, Blender було обрано як основний інструмент розробки. Він поєднує можливості професійного моделювання з відкритим кодом та повною підтримкою пайплайну для WebGL. Ключові аргументи на користь Blender [31]:

1. Недеструктивний метод роботи (Non-destructive Workflow): Використання стека модифікаторів (Modifier Stack) дозволяє змінювати параметри фасок, товщину стінок корпусу чи логічні операції (Boolean) на будь-якому етапі без руйнування базової геометрії.

2. Інструменти оптимізації та «запікання» (Baking): Blender має вбудовані засоби для ретопології та перенесення деталей з високополігональної моделі на низькополігональну (Texture Baking). Це дозволяє зберегти візуальну якість дрібних механічних вузлів при мінімальному розмірі файлу, що забезпечує швидке завантаження експоната у браузері.

3. Hard Surface моделювання: Наявність спеціалізованих інструментів для роботи з жорсткими поверхнями дозволяє точно відтворювати складні переходи форм, кнопки, перемикачі та елементи стрічкопротяжного механізму.

4. Нативна підтримка glTF/GLB: Blender забезпечує прямий експорт у формат glTF 2.0, який є стандартом для Three.js та WebGL, гарантуючи коректну передачу геометрії, анімації та PBR-матеріалів.

Таблиця 2.1

Порівняльна характеристика інструментарію

Критерій порівняння	Blender	Autodesk Maya	Cinema 4D	CAD-системи
Hard Surface	Висока	Середня	Низька	Максимальна
Топології для Web	Висока (Clean Mesh)	Висока	Середня	Низька (NURBS триангуляція)
Baking/Retopo	Вбудовані інструменти	Потребує налаштувань	Обмежені можливості	Відсутні
Експорт у glTF/Web	Нативна, стабільна	Через плагіни	Потребує налаштувань	Через проміжні формати
Вартість / Доступність	Open Source (GPL)	Висока вартість	Середня/Висока	Висока вартість

Аналіз функціональних можливостей сучасних 3D-редакторів засвідчив, що Blender є найбільш релевантним інструментом для досягнення мети проєкту. Його переваги у створенні чистої полігональної сітки, можливості недеструктивного редагування та потужні інструменти оптимізації ресурсів («запікання» текстур) дозволяють створити високодеталізовану, але «легку» модель магнітофона. Це забезпечує баланс між історичною достовірністю експоната та технічними вимогами веб-інтеграції.

2.1.2. Технології веб-візуалізації: WebGL та бібліотеки на його основі

Сучасний веб-простір давно вийшов за межі статичних сторінок і сьогодні здатний відтворювати складні інтерактивні середовища. Основою для роботи тривимірної графіки у браузері є стандарт WebGL, і розуміння його принципів є важливим для обґрунтування архітектури майбутньої платформи.

WebGL – це кросплатформний API, який дозволяє створювати 3D-графіку безпосередньо в браузері, використовуючи лише засоби HTML5, зокрема елемент `<canvas>`. Технологія базується на OpenGL ES – спрощеній версії графічного стандарту для мобільних і вбудованих систем – і відкриває JavaScript доступ до апаратних можливостей відеокарти [32]. Завдяки цьому WebGL може рендерити складні сцени з високою продуктивністю.

Графіка у WebGL будується через програмований рендеринговий конвеєр. Його робота зводиться до двох основних етапів: вершинних шейдерів, які обробляють положення вершин моделі у просторі, та фрагментних шейдерів, що визначають колір і вигляд кожного пікселя. Обидва етапи програмуються на мові GLSL, яка за синтаксисом близька до C++, і виконуються паралельно на GPU. Це дозволяє відтворювати складні матеріали, освітлення й динамічні ефекти у режимі реального часу.

Попри високу продуктивність, WebGL залишається низькорівневим інструментом. Він оперує базовими графічними примітивами й не має вбудованих понять сцени, камери чи джерел світла. Навіть для простого 3D-об'єкта розробник

має самостійно створити й керувати буферами, написати шейдери, виконати всі матричні перетворення та побудувати цикл рендерингу. Це значно збільшує обсяг коду й підвищує поріг входження: ініціалізація одного куба може зайняти понад сотню рядків. Така специфіка робить пряме використання WebGL малопрактичним для проєктів, де важливими є швидкість розробки, гнучкість та можливість швидко масштабувати функціонал.

Саме через складність WebGL виникла екосистема бібліотек, які спрощують взаємодію з ним. Вони беруть на себе ініціалізацію контексту, компіляцію шейдерів, математичні операції та структурування сцени. Three.js, Babylon.js та інші подібні рішення фактично виступають «посередниками» між WebGL і кодом програми, дозволяючи працювати зі зрозумілими об'єктами – сценами, камерами, джерелами світла чи моделями. Такі бібліотеки перетворюють сирий API на інструмент, придатний для швидкого створення інтерактивної графіки й зручної інтеграції з іншими вебтехнологіями. Це робить їх оптимальним вибором для розробки навчальних платформ та цифрових музейних експонатів.

WebGL залишається ключовим стандартом для рендерингу 3D-графіки у браузері завдяки апаратному прискоренню та широкій підтримці. Проте його низькорівнева природа ускладнює пряме використання у проєктах із короткими термінами розробки. Тому доцільним є використання високорівневої бібліотеки, яка абстрагує складність WebGL.

2.1.3. Формати обміну 3D-даними та вибір glTF/GLB

Ефективне відтворення 3D-моделі у вебсередовищі значною мірою залежить від того, у якому форматі передається модель. Навіть ідеально створена у Blender сцена може працювати повільно, якщо файл занадто важкий або якщо браузер витрачає багато часу на його розбір. Тому перед інтеграцією 3D-експоната важливо обрати формат, який поєднує компактність, швидкість завантаження та повну підтримку сучасних графічних можливостей. У цьому підрозділі розглянуто головні

формати обміну 3D-даними – OBJ, FBX та glTF – і пояснено, чому для вебплатформи музею обрано саме glTF/GLB.

OBJ протягом багатьох років був одним із найпоширеніших форматів у 3D-графіці. Він зберігає геометрію, нормалі й UV-розгортки, але практично не працює з матеріалами та зовсім не підтримує анімацію чи складну структуру сцени. Оскільки OBJ має текстовий формат, його файли зазвичай досить великі, а завантаження у браузері – повільне [33].

FBX – це більш сучасний та функціональний формат, який активно використовується в ігрових рушіях і кіновиробництві. Він підтримує анімацію, камери, освітлення та багато додаткових параметрів. Проте FBX є пропрієтарним і важким у парсингу. Для вебдодатків це створює проблему: великий файл та складна структура можуть серйозно знизити продуктивність і викликати затримки при завантаженні.

Формат glTF, розроблений Khronos Group, створений спеціально для передачі 3D-сцен у вебі. Його часто описують як «JPEG для 3D-графіки», оскільки він орієнтований на швидке та ефективне відтворення, а не на редагування [34].

Однією з ключових переваг glTF є його компактність: він підтримує сучасні методи стиснення, такі як Draco, які дозволяють суттєво зменшити вагу файлів без видимої втрати якості. Це особливо важливо для моделі «Маяка-140», яка містить багато дрібних механічних деталей.

Ще однією важливою властивістю є швидкість завантаження. Структура glTF безпосередньо відповідає тому, як WebGL зберігає дані у пам'яті GPU, тому браузеру не потрібно виконувати складні перетворення. Модель завантажується й відображається практично миттєво.

Формат також підтримує PBR-матеріали (Physically Based Rendering), що дозволяє коректно відтворити металеві панелі, пластикові кнопки та скляні поверхні індикаторів магнітофона. Крім того, glTF зберігає ієрархію об'єктів та анімацію, що критично для інтерактивного експоната: кожен рухомий елемент моделі може бути окремим вузлом у сцені [34].

Для цього проєкту обрано бінарну версію glTF – формат .glb. Він пакує всю інформацію про модель (геометрію, матеріали, текстури й анімації) в один файл. Це суттєво спрощує загальну структуру застосунку та зменшує кількість HTTP-запитів, що вигідно впливає на швидкість завантаження сцени.

У середовищі Three.js файл .glb можна імпортувати за допомогою стандартного завантажувача GLTFLoader буквально кількома рядками коду. Це робить процес інтеграції передбачуваним і зручним.

Формат glTF/GLB є оптимальним рішенням для інтерактивного музейного експоната. Він забезпечує баланс між високою якістю візуалізації, швидким завантаженням і зручністю інтеграції. Саме ці характеристики дозволяють зробити роботу з моделлю магнітофона «Маяк-140» комфортною та стабільною навіть на мобільних пристроях, що є ключовою вимогою для сучасної вебплатформи.

2.2.Інтерактивність у цифровому музейному експонаті

Інтерактивність – це не просто можливість натиснути кнопку, це фундаментальний механізм, який перетворює статичну картинку на повноцінний цифровий досвід. У цьому розділі ми проаналізуємо, як саме технології забезпечують цей діалог між користувачем і системою, та чому для нашого завдання обрано конкретний набір інструментів. Наша мета – створити середовище, яке реагує на дії відвідувача миттєво, стабільно та передбачувано, імітуючи роботу реального фізичного приладу. Важливо, щоб кожна взаємодія давала зрозумілий зворотний зв'язок: підсвічування елементів, зміна станів, плавні переходи та коректна анімація мають працювати узгоджено, без затримок і «випадкової» поведінки. Основою для створення гармонійного та естетично привабливого інтерфейсу та 3D-сцени слугують фундаментальні принципи композиції, рисунку та перспективи, викладені у навчальних посібниках НАУ. Дотримання законів композиції, зокрема використання принципів золотого перетину та контрасту, забезпечує цілісність сприйняття візуального контенту платформи та допомагає керувати увагою користувача, виділяючи ключові органи керування і смислові акценти [46].

2.2.1. Огляд підходів до інтерактивності та вибір оптимальної технології (Unity, Unreal Engine, Babylon.js, Three.js)

Створення інтерактивного музейного експоната потребує вибору такого інструменту, який забезпечить стабільну роботу у браузері, високу швидкість рендерингу та можливість точно керувати окремими частинами моделі. Ринок пропонує широкий спектр технологій для побудови 3D-взаємодії – від важких ігрових рушіїв до легковагових бібліотек поверх WebGL. У цьому підрозділі розглянуто ключові варіанти та пояснено, чому для проєкту обрано веборієнтований підхід із використанням Three.js.

Unity та Unreal Engine – це потужні платформи, розраховані на створення повноцінних 3D-додатків і відеоігор. Вони мають величезний набір інструментів: фізику, анімаційні системи, складні матеріали, інструменти побудови інтерфейсу та розвинуті редактори сцен. Однак для вебплатформи музейного типу ці можливості виявляються надлишковими.

Проблема полягає у способі доставки такого контенту. Хоча Unity та Unreal підтримують WebGL-експорт, згенеровані збірки мають значний розмір, споживають багато оперативної пам'яті й завантажуються повільно, особливо на мобільних пристроях. Крім того, інтеграція з HTML та DOM-елементами обмежена, що ускладнює поєднання 3D-сцени з вебінтерфейсом у вигляді підказок, кнопок або пояснювальних панелей.

Для задачі демонстрації одного технічного експоната такі рушії є надто важкими й не забезпечують необхідної гнучкості у взаємодії з браузером.

Babylon.js – це потужний рушій на основі WebGL, який підтримує фізику, системи частинок, PBR-матеріали та роботу з VR/AR. Він пропонує більш структурований підхід до створення сцени порівняно з Three.js і орієнтується на складні інтерактивні проєкти.

У цьому є й недоліки: Babylon.js має важчу архітектуру та більший розмір бандлу, що не завжди виправдано для музейної платформи, яка фокусується на одному об'єкті, а не на розгалуженому світі або ігровій логіці. Для нашого проєкту

важливішими є швидке завантаження й повна інтеграція з вебінтерфейсом, а не потужна фізика чи складні сцени.

Three.js – це високорівнева бібліотека, яка працює поверх WebGL і надає гнучкий об'єктно-орієнтований інтерфейс для створення 3D-сцен. Її головна перевага – легкість та прозорість. Бібліотека не приховує WebGL повністю, але абстрагує найскладніші частини: роботу зі шейдерами, створення буферів, обробку матеріалів і світла. При цьому вона залишається достатньо «легкою», щоб швидко працювати у браузері навіть на мобільних пристроях.

Three.js добре інтегрується з glTF/GLB – форматом, що використовується для моделі «Маяка-140», – і дозволяє отримати доступ до кожного вузла сцени. Це означає, що кнопки, котушки чи стрічковий механізм можна анімувати окремо і прив'язувати до дій користувача або аудіосигналу.

Крім того, Three.js зберігає повну сумісність із HTML, CSS та DOM-подіями. Це робить його зручним для реалізації гібридного вебінтерфейсу, де частина підказок чи інструментів керування розміщується поза 3D-сценою.

2.2.2. Three.js як інструмент взаємодії з моделлю

Після вибору технологічного підходу логічним кроком є визначення того, яким саме чином реалізується взаємодія користувача з 3D-моделлю. Three.js у цьому контексті виступає не просто бібліотекою для відтворення сцени, а центральним інструментом, який дозволяє будувати логіку поведінки окремих елементів експоната та поєднувати її з інтерфейсом вебплатформи.

Однією з головних переваг Three.js є його прямий доступ до структури glTF/GLB-моделі [35]. Після завантаження сцени кожен елемент – від котушки до кнопки – представлений як окремий об'єкт, до якого можна звертатися через код. Це дає змогу створювати взаємодії будь-якої складності: натискання кнопок, зміна положення перемикачів, запуск анімації стрічки або обертання механізмів під час відтворення аудіо.

Three.js забезпечує повний доступ до дерева вузлів (Scene Graph). Завдяки цьому можна:

- отримувати окремі частини моделі за назвами, заданими у Blender;
- змінювати їх положення, обертання чи масштаб у реальному часі;
- підключати поведінку об'єктів до подій користувача (клік, наведення, drag);
- створювати складні анімаційні послідовності.

Для магнітофона «Маяк-140» це означає можливість моделювати роботу механіки: підйом кнопок, рух важелів, обертання котушок та підсвічування VFD-індикаторів, які реагують на аудіосигнал.

На відміну від ігрових рушіїв, Three.js працює прямо у браузері та легко взаємодіє з HTML, CSS і DOM-подіями. Це дозволяє реалізовувати гібридний підхід: частина елементів інтерфейсу може бути у 3D-сцені, а частина – у вигляді звичайних HTML-компонентів.

Наприклад, підказки, контролери режимів або освітні блоки можна розмістити над сценою у вигляді статичних HTML-елементів, тоді як кнопки та перемикачі магнітофона залишаються інтерактивними всередині 3D-моделі. Це значно спрощує користувацьку взаємодію та зменшує поріг входу для нового відвідувача.

Three.js має вбудовану систему анімації, яка читає ключові кадри безпосередньо з glTF-файлу. Це дозволяє переносити з Blender не лише геометрію, а й підготовлені рухи елементів: натискання кнопки, коливання індикатора або фіксовані механічні послідовності.

У випадку з магнітофоном це дає змогу зробити взаємодію природнішою та ближчою до реального об'єкта. Користувач не просто бачить модель – він взаємодіє з нею так, як із фізичним пристроєм.

Three.js оптимізований для роботи у WebGL і добре масштабується під можливості різних платформ. Завдяки цьому сцена стабільно працює як на настільних ПК, так і на мобільних браузерах, де ресурси суттєво обмежені. Бібліотека автоматично використовує апаратне прискорення, а також дозволяє вручну контролювати рівень деталізації (LOD), кількість світла, тіні та інші параметри

продуктивності. Це важливо для музейного експоната, який має залишатися доступним навіть на слабких пристроях або у нестабільних мережевих умовах.

2.2.3. Реалізація аудіоінтерактивності за допомогою Web Audio API

Оскільки магнітофон є аудіопристроєм, важливо не лише відтворити його зовнішній вигляд, а й передати логіку роботи через звук. Саме тому інтерактивність моделі має реагувати на аудіосигнал у реальному часі: котушки повинні обертатися синхронно з відтворенням, а індикатори – відображати рівень сигналу. Для цього у вебсередовищі використовується Web Audio API – потужний інструмент для роботи зі звуком, що надає доступ до потокової обробки аудіо прямо в браузері [36].

Web Audio API працює на основі модульної графової архітектури. Це означає, що розробник може будувати «ланцюги обробки», підключаючи до них різні вузли: джерела звуку, фільтри, аналізатори та ефекти. У контексті музейного експоната ключовим елементом є `AnalyserNode` – вузол, який дозволяє отримувати спектральні та хвильові дані аудіосигналу в режимі реального часу. Саме ці дані використовуються для анімації індикаторів, еквайзера та синхронізації руху стрічки.

Принцип роботи такий: завантажений аудіофайл підключається до аудіоконтексту браузера, після чого пропускається через `AnalyserNode`. Далі `Three.js` отримує масив значень амплітуди або частоти, які оновлюються щок cadрово. Ці дані можна використовувати для керування будь-якими параметрами моделі – наприклад, змінювати інтенсивність світіння індикаторів або швидкість обертання котушок пропорційно до рівня сигналу. Завдяки цьому модель «оживає», і взаємодія набуває реалістичного характеру.

Web Audio API дозволяє також реалізувати синхронізацію анімації зі звуком. Через точний доступ до позиції відтворення (`currentTime`) можна синхронізувати, наприклад, початок руху стрічки чи зміни стану кнопок. Це важливо для моделювання реальної поведінки магнітофона, де механіка завжди взаємопов'язана зі звуковим процесом.

Ще однією перевагою API є його висока продуктивність: обчислення виконуються окремо від головного потоку JavaScript (у спеціалізованому аудіопотоці), завдяки чому інтерфейс та 3D-сцена не «просідають» навіть під час складної графічної взаємодії. Це дозволяє одночасно виконувати обробку звуку, рендеринг 3D-графіки та оновлення анімацій.

У підсумку Web Audio API забезпечує гнучкий та продуктивний спосіб поєднати звук і 3D-візуалізацію. Він дозволяє зробити взаємодію з магнітофоном не лише візуальною, а й поведінковою – коли кожен механічний елемент реагує на реальний аудіосигнал. Саме це створює ефект «живого» експоната та підвищує освітню цінність платформи.

2.3.Інтеграція 3D-моделі у вебплатформу

Після вибору технологічного стеку та методів реалізації інтерактивності наступним етапом є інтеграція 3D-моделі магнітофона у вебдодаток. На цьому етапі важливо не лише «вивести» модель у браузер, а й побудувати архітектуру, яка забезпечить стабільну роботу всієї системи: від інтерфейсу до обробки аудіосигналу. Тому в цьому розділі розглядається структура вебдодатка, логіка взаємодії між компонентами та ключові прийоми оптимізації, необхідні для плавного відтворення сцени навіть на малопотужних пристроях. Такий підхід дозволяє перетворити 3D-модель на повноцінний інтерактивний експонат, доступний у звичайному браузері.

2.3.1.Архітектура вебдодатка та обґрунтування вибору SPA

Інтерактивна модель магнітофона «Маяк-140» – це не просто елемент сторінки, а самостійний цифровий експонат, який постійно обробляє дані, оновлює анімації та реагує на дії користувача. Тому архітектура вебдодатка має забезпечувати безперервну взаємодію без перезавантажень сторінок та без втрат продуктивності [37]. Саме через це основою проєкту було обрано підхід SPA (Single Page Application).

У форматі SPA увесь інтерфейс працює в межах однієї HTML-сторінки. Після початкового завантаження браузер підвантажує лише ті дані, які потрібні для оновлення стану інтерфейсу або 3D-сцени. Це особливо важливо для експоната, де 3D-модель, UI-панель керування та аудіопроцесор повинні працювати синхронно в реальному часі. Перезавантаження сторінки або повний рендер DOM при кожній взаємодії негативно вплинули б на стабільність сцени та сприйняття користувачем.

Архітектурно SPA дозволяє розділити логіку додатка на окремі модулі – «відповідальності», які працюють паралельно. Наприклад, 3D-рендеринг виконується в одному циклі (`requestAnimationFrame`), а UI оновлюється у відповідь на події користувача, не блокуючи графічний потік. Окремо працює аудіопотік Web Audio API, який не залежить від FPS сцени. Така модульність мінімізує ризик «просідань» при взаємодії з моделлю.

Крім того, SPA дає можливість легко реалізувати динамічні стани: перехід між режимами («онбординг», «вільне дослідження», «режим еквалайзера»), підсвітку активних елементів, виклик інформаційних довідок – усе це працює без перезавантажень і миттєво реагує на дії користувача. Це безпосередньо впливає на зручність взаємодії з експонатом та створює ефект «живого» цифрового пристрою.

Також підхід SPA забезпечує кращу продуктивність на мобільних пристроях, оскільки основна логіка та ресурси завантажуються один раз і далі повторно використовуються. З огляду на те, що сучасні музейні вебплатформи орієнтовані на широку аудиторію, включно з мобільною, цей фактор є критично важливим.

Таким чином, архітектура SPA була обрана через її здатність забезпечити плавну, безперервну взаємодію, гнучку роботу модулів та високий рівень продуктивності – усе це напряду впливає на якість роботи інтерактивного експоната.

2.3.2. Взаємодія компонентів: 3D-модель, UI та аудіопроцесор

Щоб інтерактивна модель магнітофона працювала як цілісний експонат, її компоненти мають бути з'єднані в єдину систему. У вебплатформі кожен елемент – 3D-сцена, інтерфейс керування та аудіопроцесор Web Audio API – виконує свою роль,

але ефект «живої» взаємодії з'являється лише тоді, коли вони синхронізовані між собою. Тому важливо не просто вивести модель у браузер, а побудувати зв'язки між модулями так, щоб будь-яка дія користувача миттєво впливала на стан 3D-об'єкта і навпаки.

У центрі всієї логіки знаходиться `Rendering Loop` – основний цикл, у якому `Three.js` оновлює сцену й перевіряє стан елементів. Саме в цьому циклі зчитуються дані з аудіоаналізатора, відстежується натискання кнопок інтерфейсу та оновлюються параметри 3D-моделі. Наприклад, коли користувач натискає кнопку «Play» у UI, ця подія викликає запуск аудіотреку через `Web Audio API`, а цикл рендерингу «підхоплює» новий стан і запускає обертання катушок на моделі.

Аудіопроектор `Web Audio API` працює незалежно від графічного потоку, але передає дані про амплітуди та частоти через масиви, які оновлюються в реальному часі. Ці масиви використовуються `Three.js` для оновлення індикаторів: наприклад, анімація еквайзера базується на значеннях з `AnalyserNode`. Таким чином, звук і 3D-графіка залишаються синхронізованими незалежно від FPS сцени.

UI взаємодіє з моделлю через подієвий механізм. Кожна кнопка, перемикач або повзунок HTML має «слухач» подій, який передає команду в модуль логіки. Далі ця команда перетворюється на конкретну дію: зміну стану кнопки на 3D-моделі, оновлення анімації або зміну конфігурації аудіографа. Наприклад, під час натискання на 3D-кнопку «Стоп» у `Three.js`, відповідний обробник зупиняє `BufferSource` у `Web Audio API` і зупиняє рух стрічки на моделі.

Взаємодія компонентів відбувається у форматі мікроподій, де кожен модуль відповідає лише за свій сегмент логіки, але всі вони обмінюються станами через централізований контролер. Така структура дозволяє легко розширювати платформу: додавати нові режими роботи індикаторів, інтегрувати панелі підказок або розширювати механіку керування без ризику порушити баланс між існуючими компонентами.

У результаті вся система працює як злагоджений механізм: UI передає команди, аудіопроектор генерує динамічні дані, а `Three.js` відповідає за візуальну частину. Завдяки цьому користувач отримує цілісний досвід взаємодії, де кожен елемент

магнітофона реагує природно та узгоджено, що значно підсилює ефект реалістичного відтворення роботи пристрою.

2.3.3. Оптимізація продуктивності у браузері

Інтерактивна 3D-сцена у браузері завжди працює в умовах обмежених ресурсів. Користувач може запускати експонат на слабкому ноутбуці, телефоні чи планшеті, а браузер ділитиме процесор і відеокарту між десятками вкладок. Тому оптимізація продуктивності – не просто «приємний бонус», а обов’язкова умова, щоб модель магнітофона працювала плавно та стабільно. У цьому підрозділі розглядаються ключові техніки, які дозволяють зменшити навантаження на GPU й забезпечити стабільний FPS.

Першим кроком є оптимізація самої 3D-моделі. Для вебу критично важливо мінімізувати кількість полігонів і прибрати зайву деталізацію, яку все одно не буде видно в інтерактивному перегляді. У процесі підготовки моделі «Маяка-140» застосовувалися ретопологія, зменшення кількості дрібних елементів і об’єднання мешів, щоб скоротити кількість окремих об’єктів у сцені. Це позитивно впливає як на швидкість завантаження, так і на продуктивність під час рендерингу.

Другим важливим напрямом є робота з матеріалами та текстурами. Веб-рендеринг найкраще працює тоді, коли використовується невелика кількість текстур стандартних розмірів (512–2048 px) та форматів зі стисненням (JPEG, WebP або спеціалізовані формати для glTF). Формат GLB дозволяє вбудовувати матеріали з PBR-параметрами, але надмірне використання складних шейдерів може знизити FPS, тому в роботі застосовувалися максимально прості PBR-матеріали без додаткових кастомних шейдерів.

Ще однією важливою технікою є оптимізація рендер-циклу Three.js. Rendering Loop працює на основі requestAnimationFrame, тому будь-яка зайва операція в циклі безпосередньо впливає на продуктивність. Важливо розділяти оновлення, які мають виконуватися кожного кадру (анімація котушок, реакція індикаторів), і ті, що можна оновлювати рідше (стан UI, допоміжні обчислення). Для частини розрахунків

використовуються кешовані значення, щоб мінімізувати повторні матричні та геометричні операції.

Оптимізації також потребує взаємодія з Web Audio API. Попри те, що аудіопотік працює в окремому потокові браузері, передавання великих масивів спектральних даних може створювати зайве навантаження. Тому замість зчитування повного частотного спектру використовуються лише ті частоти, які потрібні для візуалізації індикаторів. Це дозволяє зменшити кількість обчислень без втрати якості взаємодії.

Важливим елементом є адаптація продуктивності під різні пристрої. Three.js дозволяє динамічно знижувати якість рендерингу – наприклад, регулювати розмір тіней, вимикати постобробку або зменшувати роздільну здатність рендера для слабких пристроїв [38]. Такий адаптивний підхід гарантує, що модель працюватиме стабільно навіть на смартфонах середнього класу.

У комплексі ці техніки забезпечують плавну та передбачувану роботу інтерактивної моделі в браузері. Користувач отримує чітку анімацію, швидку реакцію інтерфейсу та реалістичну поведінку моделі, а платформа залишається доступною на широкому спектрі пристроїв – що є ключовою вимогою сучасних музейних вебекспонатів.

2.4.Методика створення 3D-моделі магнітофона «Маяк-140»

Створення точної та водночас оптимізованої 3D-моделі – один із ключових етапів побудови інтерактивного музейного експоната. У випадку магнітофона «Маяк-140» важливо не лише відтворити загальний вигляд пристрою, а й передати логіку його механіки: рух котушок, поведінку стрічки, роботу кнопок і візуальні властивості матеріалів корпусу. Тому процес моделювання складається з трьох взаємопов'язаних частин: декомпозиції пристрою на функціональні вузли, підготовки матеріалів для коректного PBR-рендерингу та оптимізації моделі для експорту у формат GLB. У цьому підрозділі докладно описано кожен етап – від первинного аналізу конструкції до її інтеграції у вебсередовище.

2.4.1. Декомпозиція пристрою та моделювання ключових вузлів

Перше завдання під час створення 3D-моделі – правильно розкласти реальний пристрій на логічні складові. Магнітофон «Маяк-140» – це не монолітний об’єкт, а набір механічних вузлів, які працюють по-різному й вимагають різного підходу до моделювання [39]. Декомпозиція дозволяє точно визначити, які елементи будуть статичними, а які – інтерактивними та анімованими. Це критично для подальшої роботи у Three.js, де кожен вузол має свою точку обертання, позицію та поведінкову логіку.

У першу чергу було виділено три основні групи елементів: корпус, механічні частини стрічкопротяжного механізму та елементи керування. Корпус моделюється як цілісний об’єкт із чітко вирізаними отворами під кнопки, індикатори та панелі. На цьому етапі важливо зберегти правильні пропорції, тому для моделювання використовувалися фотографії з орто-проекцією та вимірювання реального пристрою.

Другий етап – моделювання механічних вузлів. Це найбільш складна частина, оскільки котушки, притискні ролики, напрямні стрічки та маховики мають власну кінематику. Усі ці об’єкти моделювалися окремо, з урахуванням їхньої реальної форми та взаєморозташування. Особливу увагу приділено центральному блоку зі стрічкопротяжним механізмом: його геометрія впливає на достовірність усієї анімації.

Елементи керування – кнопки, повзунки еквалайзера, перемикачі – моделювалися як окремі об’єкти з власними pivot-точками. Це дозволяє прив’язати їхнє переміщення до станів UI у браузері. Наприклад, кнопки мають мати мінімальну глибину натиску і працювати без спотворень; повзунки повинні рухатися строго по своїх напрямних.

Завершальним кроком є об’єднання вузлів у єдину ієрархічну структуру. У Blender формується дерево об’єктів, яке згодом буде прочитане Three.js у форматі

Scene Graph. Це дозволяє керувати кожним елементом окремо та забезпечує коректну поведінку моделі в інтерактивному середовищі.

2.4.2. Підготовка матеріалів і освітлення для PBR-рендерингу

Щоб 3D-модель магнітофона «Маяк-140» виглядала реалістично у браузері, недостатньо створити правильну геометрію – необхідно коректно налаштувати матеріали та освітлення. У вебсередовищі для цього використовується підхід PBR (Physically Based Rendering) [40], який дозволяє досягти фізично правдивої поведінки світла на поверхнях. Це особливо важливо в технічних експонатах, де велику роль відіграють металеві деталі, пластик, скло індикаторів і текстуровані поверхні корпусу.

Першим етапом є визначення матеріальних зон на моделі. Магнітофон складається з кількох видів поверхонь: металеві панелі корпусу, пластикові кнопки й перемикачі, гумові ролики стрічкопротяжного механізму, напівпрозорі елементи індикаторів. Для кожного типу матеріалу створюється окремий PBR-набір, що включає базову карту (Base Color), карту шорсткості (Roughness), карту металевості (Metallic) та, за потреби, карту нормалей (Normal Map). Такий поділ дозволяє досягнути гармонійного балансу між деталізацією та продуктивністю.

Для металевих поверхонь корпусу важливо правильно передати властивості холодного алюмінію – з легким відблиском, але без дзеркального блиску. Тому Metallic значення встановлюється у середньому діапазоні, а Roughness підбирається таким чином, щоб поверхня виглядала матовою, але «живою». Пластикові елементи, навпаки, потребують більших значень шорсткості, щоб уникнути надмірних відблисків у сцені. Для кнопок і перемикачів використовуються більш прості матеріали без карт нормалей, оскільки дрібні нерівності на них не помітні в інтерактивному режимі.

Окремим етапом налаштовується освітлення. У сцені використовується одна HDRI-текстура для отримання коректних рефлексів на матеріалах і додаткове м'яке ключове світло, що підкреслює форму об'єкта. HDRI дозволяє зберегти реалістичну

поведінку відблисків на металічних поверхнях і водночас не перевантажує рендеринг. Додатково застосовується слабке заповнювальне світло, щоб уникнути різких тіней, які можуть утруднити сприйняття моделі.

Для індикаторів (VFD або LED-елементи) використовується емісивний канал матеріалу. Це дозволяє створити природне підсвічування, інтенсивність якого в подальшому контролюється через Three.js відповідно до даних з аудіоаналізатора. Таким чином, модель не лише виглядає реалістично, а й демонструє поведінку справжнього електронного індикатора.

Усі матеріали готуються з урахуванням майбутнього експорту у формат GLB. Це означає суворе дотримання стандартів glTF 2.0: використання PBR-матеріалів, оптимізованих текстур та відсутність нестандартних шейдерів. Такий підхід гарантує, що модель виглядатиме однаково в Blender і у веббраузері, без додаткових налаштувань або корекції.

2.4.3. Оптимізація й експорт моделі у форматі GLB

Після завершення моделювання та налаштування матеріалів наступним важливим етапом є підготовка моделі до використання у вебсередовищі. Формат GLB (бінарна версія glTF 2.0) був обраний як основний вихідний формат, тому що він забезпечує швидке завантаження у браузері, коректну передачу PBR-матеріалів та компактний розмір файлу. Щоб модель магнітофона «Маяк-140» працювала стабільно на широкому спектрі пристроїв, перед експортом необхідно виконати низку оптимізацій.

Першим етапом є зменшення полігонального навантаження. Blender дозволяє застосовувати ретопологію, модифікатор Decimate або ручне спрощення вторинних елементів. При цьому важливо зберегти загальну форму й правильну структуру ключових вузлів – котушок, кнопок, індикаторів – адже саме вони беруть участь в анімації. Статичні деталі корпусу можуть бути спрощені сильно, оскільки користувач рідко розглядає їх зблизька. Такий підхід дає змогу суттєво зменшити навантаження на GPU під час рендерингу.

Другий етап – оптимізація текстур. GLB працює найефективніше, коли розміри текстур знаходяться в діапазоні степенів двійки (512, 1024, 2048 px). Це прискорює завантаження і покращує роботу з пам'яттю відеокарти. Текстури зберігаються у форматах JPEG або WebP для BaseColor і PNG для карт нормалей. Важливим кроком є запікання (baking) дрібних деталей у текстури – це дозволяє передати складні поверхні без високої геометричної деталізації.

Після цього готується структура сцени. GLB підтримує ієрархію об'єктів, анімації та окремі вузли, тому важливо, щоб модель мала правильно налаштовані pivot-точки та логічну структуру. Усі об'єкти, які будуть анімовані в Three.js, повинні мати унікальні назви та бути згрупованими в окремі вузли: наприклад, LeftReel, RightReel, PlayButton, StopButton. Це полегшує програмну взаємодію після експорту.

Фінальним кроком є експорт у формат GLB. Blender надає стандартний glTF 2.0 експортер, який дозволяє зберегти геометрію, матеріали, UV-розгортки, емісію, структуру сцени та анімації. Важливо активувати такі параметри, як:

- “Apply Modifiers” – щоб модель експортувалася зі всіма фінальними геометричними змінами;
- “Embed Textures” – щоб вся сцена була упакована в один файл .glb, що значно скорочує кількість HTTP-запитів;
- “Animation” – якщо передбачена експортована анімація катушок або кнопок.

Після експорту модель проходить перевірку у візуалізаторі glTF Viewer (Khronos, Don McCurdy), де можна оцінити правильність відображення матеріалів, коректність PBR-параметрів (металевість/шорсткість), відповідність текстур, поведінку анімацій та загальну продуктивність сцени [41]. Окремо перевіряється, чи правильно працюють ієрархія об'єктів, точки обертання, кліпи анімацій та їх відтворення без ривків або зміщень. Це дозволяє виявити можливі помилки ще до інтеграції у Three.js, швидко локалізувати проблемне місце (матеріал, UV, анімація або світло) та уникнути непотрібних повторних експортів і втрати часу на «відладку в браузері».

У підсумку оптимізована та коректно експортована модель у форматі GLB забезпечує швидке завантаження, ефективну роботу рендеру та точне відтворення матеріалів у браузері. Завдяки цьому зменшується навантаження на GPU/CPU, покращується стабільність FPS і підвищується передбачуваність поведінки 3D-об'єкта на різних пристроях. Це створює надійну основу для інтерактивної презентації «Маяка-140» у вебплатформі та подальшого додавання взаємодії (активні елементи, підказки, керування станами та анімаційні реакції).

2.5.Проектування користувацького інтерфейсу (UI/UX)

Якість взаємодії з цифровим музейним експонатом визначається не лише тим, наскільки точно змодельовано об'єкт, а й тим, наскільки зручно користувач може з ним працювати. У нашому випадку важливо пам'ятати: відвідувач взаємодіє з «Маяком-140» не через фізичні кнопки та не через VR-контролери, а через звичайний екран – ноутбука, смартфона чи планшета. Це накладає свої обмеження й формує специфічні вимоги до UI/UX.

Саме тому у вебплатформі доводиться створювати додатковий інтерфейс, який доповнює модель і компенсує відсутність фізичного контакту з пристроєм. Деякі дії, які у реальному магнітофоні виконуються апаратно, потрібно винести в окремі UI-компоненти – наприклад, вибір музичного треку, перемикання режимів демонстрації, доступ до підказок або активація освітніх модулів. Завдання інтерфейсу – дати користувачеві всі необхідні інструменти, але без перевантаження сцени та без втрати фокусу на самому експонаті.

Навчальні музейні платформи загалом повинні пояснювати складні речі просто, тому дизайн інтерфейсу має бути мінімалістичним, передбачуваним та адаптивним. У цьому розділі розглянуто ключові принципи проектування UI, вимоги доступності (WCAG 2.1) та сценарії взаємодії, які дозволяють зробити використання вебплатформи інтуїтивним і захопливим, навіть коли взаємодія відбувається через стандартний екран.

2.5.1. Принципи дизайну для навчальних музейних платформ

Інтерфейс цифрового музейного експоната має допомагати користувачеві розуміти складні технічні процеси, але не відволікати від самої моделі. Оскільки взаємодія з магнітофоном «Маяк-140» відбувається через звичайні екрани – телефон, планшет або ноутбук – UI мусить компенсувати відсутність фізичних органів керування. У реальному пристрої користувач покладається на тактильність: натискання кнопки, фіксацію повзунка, зворотний механічний відгук. У цифровій версії цього немає, тому частину взаємодій потрібно винести в окремі інтерфейсні елементи.

Перший принцип – мінімалізм. Користувач має бачити передусім сам об'єкт, а не панель, заповнену елементами. Тому UI вводиться поступово, лише в той момент, коли це дійсно потрібно [44]. Наприклад, вибір музичного треку не вбудовується у модель, а реалізується як виносна панель або окремий блок керування над сценою. Такий підхід дозволяє не порушувати візуальну цілісність експоната й уникати перевантаження екрану.

Другий принцип – контекстність. Інтерфейс повинен «підлаштовуватися» під те, що робить користувач. Елементи, які не є актуальними у певний момент, приховуються або згортаються. Наприклад, кнопки навігації з'являються лише під час першого знайомства з моделлю, а індикатори мають підказки лише у режимі аналізу роботи еквалайзера. Усе це допомагає уникнути зайвого візуального шуму та формує природну послідовність взаємодії.

Третій принцип – послідовність і передбачуваність. Користувач повинен легко впізнавати логіку інтерфейсу: кольорові акценти, стани кнопок, поведінку підказок [42]. Якщо зелений колір означає активний стан кнопки в одному модулі, він має означати те саме і в інших. Така уніфікація полегшує навчання користувача й робить платформу зручною навіть для тих, хто вперше стикається з 3D-вебекспонатами.

Додатковим аспектом є адаптивність. Оскільки значна частина аудиторії користуватиметься платформою на смартфонах, UI повинен масштабуватися, змінювати розміщення елементів та переходити в компактніший режим без втрати

функціональності [43]. На малих екранах, наприклад, виносні панелі автоматично згортаються у бокові меню або «плаваючі» кнопки.

Сукупність цих принципів забезпечує інтерфейс, який допомагає користувачеві зосередитися на вивченні роботи магнітофона, не перевантажує його зайвою інформацією і при цьому надає всі інструменти, необхідні для взаємодії з моделлю через звичайний екран.

2.5.2. Забезпечення доступності вебплатформи (WCAG 2.1)

Доступність – один із ключових аспектів сучасних освітніх вебплатформ. Музейні сервіси відвідує широка аудиторія: користувачі з різним досвідом, різними пристроями та різними потребами. Саме тому під час розробки інтерактивної моделі магнітофона «Маяк-140» було враховано вимоги стандарту WCAG 2.1 (Web Content Accessibility Guidelines) [28], який визначає правила створення доступного цифрового контенту.

Основою стандарту є чотири принципи: сприйнятність, керованість, зрозумілість та надійність. У контексті нашої вебплатформи вони реалізовані у кількох напрямках.

Перший напрям – альтернативна навігація. Оскільки 3D-сцена може бути складною для користувачів, які не можуть взаємодіяти з мишею, було додано підтримку клавіатурної навігації. Користувач може перемикатися між інтерактивними зонами моделі за допомогою клавіші Tab, а вибір здійснюється через Enter. Це дозволяє використовувати платформу людям із обмеженою моторикою або тим, хто працює з клавіатурою за звичкою.

Другий напрям – семантичне маркування інтерфейсу. Усі кнопки, перемикачі, підказки та інформаційні панелі мають семантичні html-атрибути (aria-label, role, aria-pressed), що робить їх доступними для скрінрідерів. Наприклад, кнопка «Play» озвучується як “Кнопка: увімкнути відтворення”, а регулятори еквалайзера мають відповідні стани, які повідомляються голосовим інтерфейсом. Це важливо, оскільки

інтерфейс взаємодіє з 3D-моделлю, але користувач повинен мати доступ до його функцій незалежно від можливості бачити сцену.

Третій напрям – контрастність і читабельність. PBR-графіка часто має складне освітлення, тому для цифрових індикаторів, панелей і текстових елементів було обрано контрастні комбінації кольорів, які відповідають мінімальним вимогам WCAG (рівень AA або вище). Це гарантує, що показники VFD, підказки чи елементи інтерфейсу будуть читатися навіть користувачами зі зниженим зором.

Четвертий напрям – адаптивність для різних пристроїв. WCAG 2.1 акцентує увагу на доступності на мобільних пристроях, де особливо важлива розмірність інтерактивних зон. Тому кнопки і елементи керування мають збільшені touch-області, а бокові панелі інтерфейсу адаптивно перебудовуються в мобільний вигляд. Це дозволяє користувачеві взаємодіяти з моделлю комфортно навіть на невеликому екрані.

Окремо реалізовано підтримку “reduced motion” – системного параметра, який зменшує кількість анімацій для людей, які погано переносять рух або мають вестибулярні порушення. Якщо браузер фіксує відповідний системний параметр, анімації підказок та декоративні переходи зменшуються або вимикаються.

Завдяки цим заходам платформу можна використовувати незалежно від можливостей користувача чи типу пристрою. Це робить експонат не лише технологічним, а й справді універсальним – доступним для максимально широкої аудиторії.

2.5.3. Сценарії взаємодії користувача

Сценарії взаємодії в платформі будуються навколо простих і зрозумілих дій, які користувач виконує під час роботи з цифровою моделлю магнітофона «Маяк-140». Основна мета – забезпечити плавний досвід, де технічна складність механізму прихована за інтуїтивними діями.

1. Вибір музики через завантаження MP3 (цифрова фонотека). Початковий екран платформи працює як «цифрова фонотека»: користувач обирає MP3-файл зі

свого пристрою (або з локальної бібліотеки платформи, якщо вона передбачена). Після вибору композиції система завантажує аудіо та переводить користувача до 3D-моделі магнітофона.

2. Натискання кнопки Play: анімація касети і запуск аудіо. Після натискання кнопки Play запускаються три паралельні процеси:

- відтворюється анімація вставляння касети;
- MP3 починає програвання через аудіоплеєр платформи (Web Audio API);
- Web Audio API передає спектральні дані у VFD-індикатор, який реагує на музику в режимі реального часу.

Користувач бачить, як механіка, звук та індикатори працюють синхронно – як у справжньому «Маяку-140».

3. Кнопка «Пауза»: тимчасове зупинення без втрати стану. На відміну від кнопки Stop, Пауза не скидає стан моделі.

Після натискання:

- аудіо призупиняється без скидання позиції відтворення;
- котушки припиняють рух;
- VFD-індикатор «завмирає»;
- кнопка Play переходить у режим «Resume».

При повторному натисканні – все відновлюється з того самого місця. Це робить взаємодію природнішою для сучасного користувача, який звик до стандартної моделі Play/Pause.

4. Регулювання гучності через крутилку. Під час відтворення користувач може змінювати гучність через аналоговий поворотний регулятор.

Обертання «крутилки» напряму пов'язується з вузлом GainNode у Web Audio API, що дозволяє плавно змінювати рівень аудіо без ривків і затримок.

5. Кнопка Касета: повне завершення сеансу і повернення до вибору музики. Після натискання кнопки Касета:

- трек зупиняється;
- анімації повертаються у початковий стан;
- VFD вимикається;

– інтерфейс плавно повертає користувача на у початковий стан із можливістю вибору нової музики.

Таким чином, Касета – це «обнулення», а Pause – «тимчасова зупинка».

Разом ці сценарії формують цілісний, послідовний і зрозумілий користувацький шлях. Платформа не просто показує 3D-модель, а веде користувача через досвід – від першого знайомства до глибокого розуміння принципів роботи магнітного запису.

Висновки до розділу 2

У другому розділі було сформовано технічний фундамент для розробки інтерактивної вебплатформи, присвяченої магнітофону «Маяк-140». На основі порівняльного аналізу інструментів 3D-моделювання обґрунтовано вибір Blender як основного середовища підготовки моделі. Саме він поєднує можливості професійного hard surface-моделювання, недеструктивного редагування та оптимізації геометрії з нативною підтримкою формату glTF/GLB, що є критично важливим для подальшої інтеграції у WebGL-середовище.

Було показано, що «чистий» WebGL, попри свою продуктивність і гнучкість, є занадто низькорівневим для швидкої розробки навчального музейного експоната. Тому для рендерингу та керування сценою було обрано бібліотеку Three.js, яка абстрагує складність роботи з шейдерами та буферами і водночас забезпечує прямий доступ до структури glTF/GLB-моделі. У поєднанні з форматом GLB, оптимізованим для швидкого завантаження та коректної передачі PBR-матеріалів, це дозволило сформуванню ефективний пайплайн: «Blender → glTF/GLB → Three.js».

Окрему увагу приділено інтерактивності та роботі зі звуком. Завдяки використанню Web Audio API та AnalyserNode аудіосигнал стає не лише фоновим елементом, а частиною логіки експоната: його параметри використовуються для керування індикаторами, еквайзером та поведінкою моделі. У поєднанні з архітектурою SPA це забезпечує безперервну взаємодію без перезавантаження сторінок, синхронну роботу 3D-сцени, інтерфейсу та аудіопроектора, а також можливість адаптації продуктивності під різні пристрої.

Важливим результатом розділу стало проектування структури користувацького інтерфейсу (UI/UX) та сценаріїв взаємодії. Було визначено принципи мінімалістичного й контекстного дизайну, сформульовано вимоги до адаптивності та доступності відповідно до WCAG 2.1, а також описано ключові сценарії використання: вибір треку, запуск відтворення, керування гучністю, пауза й зупинка з поверненням до початкового екрана. Усе це дозволило перетворити 3D-модель на повноцінний освітній інструмент, орієнтований на широку аудиторію, яка взаємодіє з експонатом через звичайний веббраузер.

Таким чином, у розділі 2 послідовно обґрунтовано вибір технологій, методів моделювання та архітектурних рішень, необхідних для реалізації цифрового музейного експоната. Ці напрацювання створюють основу для наступного розділу, у якому буде детально розглянуто практичну реалізацію вебплатформи, інтеграцію 3D-моделі «Маяка-140» та розробку функціонального прототипу.

РОЗДІЛ 3

СТВОРЕННЯ ІНТЕРАКТИВНОЇ МОДЕЛІ «МАЯК-140» ТА ОЦІНКА ЕФЕКТИВНОСТІ ПРОЦЕСУ

Виконаний у попередніх розділах теоретичний аналіз і планування заклали основу для найцікавішого етапу – практичної реалізації веб-платформи. Тепер наше завдання полягає в тому, щоб перевести напрацьовані концепції у робочий цифровий продукт. У цьому розділі ми детально розглянемо процес створення інтерактивного експоната: від перших полігонів у 3D-редакторі до коду, який «оживляє» механіку магнітофона прямо в браузері.

Розробку логічно поділено на три основні етапи. Перший – робота з 3D-графікою: створення геометрії моделі, оптимізація сітки та налаштування PBR-матеріалів, щоб досягти максимально реалістичного вигляду. Другий етап – програмна частина, де модель інтегрується у веб-середовище через Three.js, а взаємодія і поведінка елементів налаштовуються за допомогою Web Audio API. Третій етап – тестування та оцінювання, які дозволяють перевірити стабільність роботи платформи, її продуктивність на різних пристроях та навчальну ефективність.

Мета цього розділу – показати, як обраний технологічний стек дає можливість візуалізувати процеси, які зазвичай лишаються невидимими, і створити наочний, інтерактивний цифровий двійник технічного пристрою.

3.1. Підготовка та імплементація оптимізованої 3D-моделі

Першим етапом практичної реалізації стало створення цифрового двійника магнітофона «Маяк-140». Основний виклик тут полягає у пошуку балансу між якістю візуалізації та технічними обмеженнями веб-середовища. Модель має виглядати максимально реалістично, але водночас залишатися «легкою» для браузера.

Іншими словами, потрібно досягти фотореалістичного результату без перевантаження сцени зайвими полігонами. Надмірно деталізована модель знижує

FPS (кількість кадрів за секунду), що негативно впливає на плавність і комфорт взаємодії. Тому на етапі моделювання важливо продумати, які елементи мають бути справді опрацьовані в 3D, а які можна відтворити за допомогою текстур, нормалей чи PBR-матеріалів. Саме цей баланс визначає, наскільки ефективно віртуальна модель працюватиме у браузері та як реалістично вона виглядатиме. Створення технічно достовірних 3D-моделей об'єктів музею (магнітофонів) вимагало застосування інженерних принципів геометричного моделювання та нарисної геометрії, що є основою для роботи у графічних середовищах. Ці методи, детально описані в навчальних посібниках НАУ, дозволили забезпечити точність розмірів, пропорцій та взаємного розташування вузлів при відтворенні техніки магнітного запису [47].

3.1.1. Створення 3D-моделі магнітофона «Маяк-140» у Blender

Для моделювання використовуємо Blender, беремо наші відзняті референси та додаємо фотографії та і технічну документації про магнітофон, яку знайшли в інтернеті. Роботу починаємо з так званого блокінгу (Greyboxing) – етапу, коли спершу вибудовуються прості форми для перевірки пропорцій і загальної геометрії корпусу. Це дозволяє ще на старті переконатися, що модель виглядатиме правильно і відповідатиме оригіналу.

Далі розпочався основний процес – створення геометрії в техніці Hard Surface Modeling. Базову форму корпусу будували за допомогою модифікаторів *Boolean* (для вирізання отворів під кнопки, індикатори та панелі) та *Bevel* (щоб додати реалістичні фаски й м'які переходи на ребрах). Оскільки булеві операції часто «ламають» топологію і створюють N-гони, проблемні ділянки довелося вручну ретопологізувати. Для покращення якості затінення також був застосований модифікатор *Weighted Normal*, який допомагає уникнути характерних артефактів на твердих поверхнях.

Не менш важливою частиною стала декомпозиція моделі на окремі об'єкти. Усі елементи, які мають бути анімовані через код – котушки, притискний ролик, кнопки ЛПМ – були відокремлені від корпусу та отримали окремі меші. Щоб їх легше було знайти у Three.js, кожному елементу призначили логічне ім'я латиницею, наприклад

btn_PLAY, *btn_POWER*, *Volume_CONTROL*. Саме цей крок забезпечує коректну взаємодію між 3D-моделлю та Javascript-логікою.

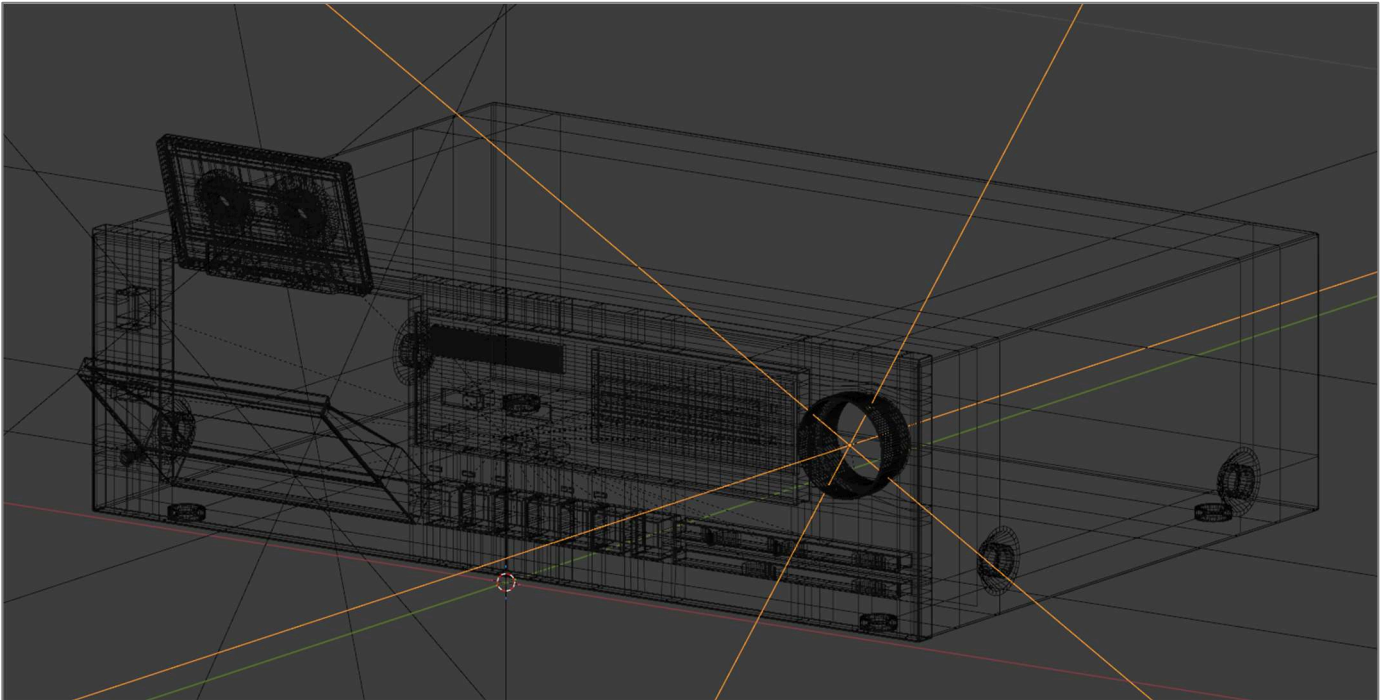


Рис. 3.1. Сітка моделі Маяк-140

Фінальним технічно важливим кроком стало налаштування *Pivot Point* для кожного рухомого елемента. Наприклад, у котушок центр обертання був точно розташований у геометричному центрі осі, щоб їх обертання під час анімації виглядало стабільно і не “плавало”. Коректний *Pivot* – основа того, щоб механіка у браузері працювала природно та без візуальних дефектів.

3.1.2. Текстурування

Текстурування 3D-моделі, призначеної для подальшого використання у веб-середовищі за допомогою бібліотеки *Three.js*, має низку специфічних особливостей, зумовлених обмеженнями продуктивності браузера та принципами роботи *WebGL*. На відміну від офлайн-рендерингу, де основний акцент робиться на максимальну візуальну достовірність, у веб-проєктах першочерговим завданням є досягнення балансу між якістю зображення та швидкодією. Текстури в цьому випадку мають

бути максимально інформативними, але водночас оптимізованими за розміром і кількістю використовуваних матеріалів.

Ключовим етапом підготовки моделі є коректна UV-розгортка. Для веб-експорту доцільно використовувати одну або мінімальну кількість UV-карт, що зменшує навантаження на відеопам'ять і спрощує роботу матеріалів у Three.js. Особливо важливо уникати перекриття UV-островів у випадках, коли застосовуються унікальні текстури, зокрема карти нормалей та емісії. Чітка й логічна UV-структура без надмірної фрагментації забезпечує стабільне відображення матеріалів у браузері та полегшує подальшу оптимізацію сцени.

Для збереження високої візуальної деталізації при мінімальній кількості полігонів у веб-проектах широко застосовується запікання (Baking) текстур. Зокрема, це стосується карт нормалей, ambient occlusion, roughness і metallic, які використовуються в PBR-матеріалах. Завдяки запіканню дрібні геометричні деталі з High-poly моделі переносяться на Low-poly версію у вигляді текстурної інформації. Такий підхід дозволяє досягти враження складної поверхні без реального збільшення геометрії, що є критично важливим для забезпечення високого FPS у браузері.

Під час підготовки моделі для вебу не рекомендується приділяти значну увагу налаштуванню освітлення та камер у середовищі Blender. У фінальній веб-сцені ці параметри не використовуються безпосередньо, оскільки освітлення, типи джерел світла, їх інтенсивність, а також положення і поведінка камери контролюються засобами Three.js на рівні програмного коду. У Blender світло та камери доцільно застосовувати виключно для попереднього перегляду та оцінки матеріалів, а не як частину фінальної візуалізації.

Для того щоб зовнішній вигляд моделі у Blender максимально відповідав її відображенню у браузері, рекомендовано використовувати рендер Eevee. Як real-time рушій, Eevee найближче відтворює принципи роботи WebGL і Three.js: спрощену модель освітлення, обмежену кількість відбиттів та поведінку PBR-матеріалів у реальному часі. На відміну від Cycles, який орієнтований на фізично точний офлайн-рендеринг, Eevee дозволяє ще на етапі підготовки оцінити реальний вигляд моделі у веб-середовищі.

Окремо слід враховувати обмеження Three.js щодо роботи з текстурами. Бібліотека не підтримує процедурні текстури та матеріали, згенеровані безпосередньо в Blender. Усі текстурні карти мають бути представлені у вигляді растрових зображень. Тому карти base color, normal, roughness, metallic, emissive та інші необхідно створювати або запікати в Blender, або допрацьовувати у графічних редакторах, зокрема в Adobe Photoshop. Такий підхід забезпечує коректний експорт у формат glTF/GLB та гарантує, що матеріали у Three.js відтворюватимуться стабільно і без втрати візуальної якості.

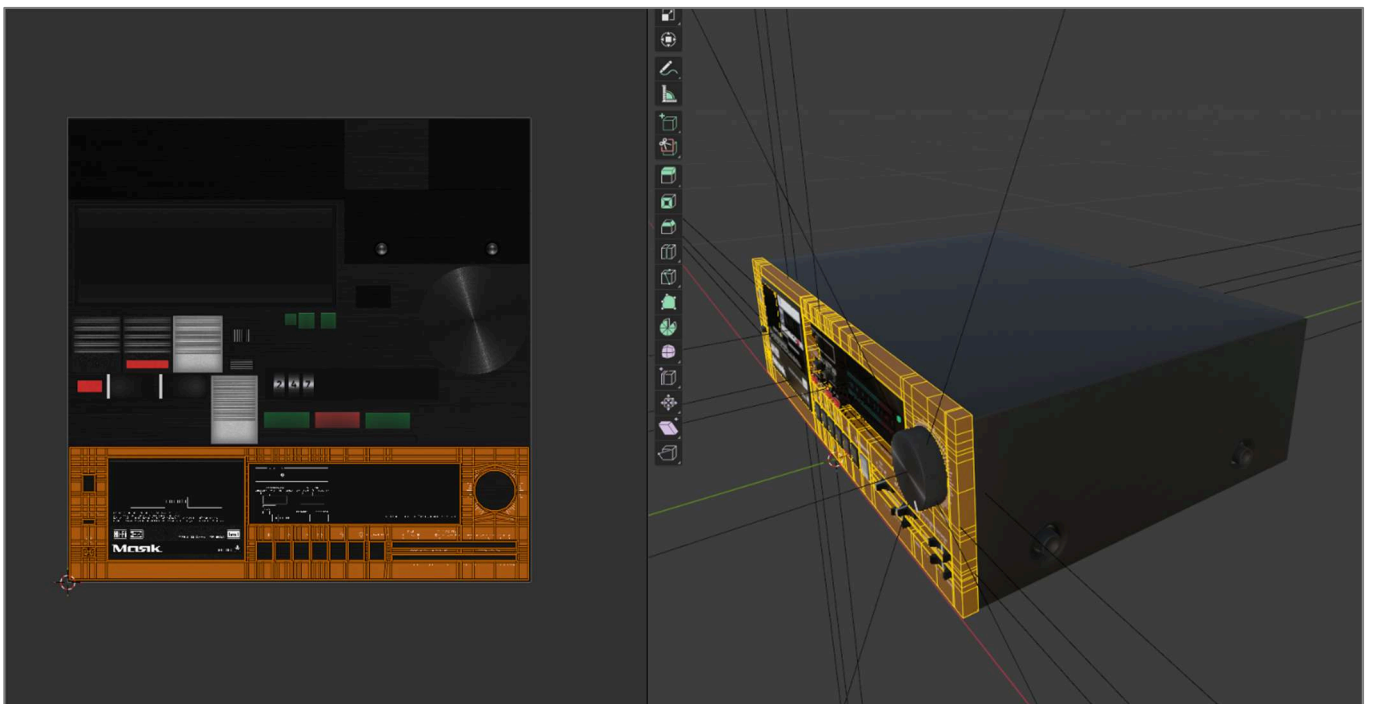


Рис. 3.2. Модель Маяк-140 з накладеними текстурами

3.1.3. Анімація за допомогою NLA, оптимізація моделі та експорт у GLB

Під час підготовки 3D-моделі для інтерактивного використання у веб-середовищі важливу роль відіграє система анімацій, оскільки саме вона визначає поведінку об'єкта у відповідь на дії користувача. Для реалізації таких анімацій у середовищі Blender доцільно використовувати NLA-треки (Non-Linear Animation). Вони дозволяють організувати анімацію не як єдину лінійну послідовність, а як набір окремих логічних кліпів, кожен з яких відповідає за конкретну дію або стан елемента: натискання кнопки, обертання котушки, вмикання індикатора тощо. Такий підхід

особливо ефективний для моделювання технічних пристроїв, де одночасно може працювати велика кількість незалежних механізмів.

Використання NLA-треків дозволяє чітко розділити статичну частину моделі та її поведінку. Кожен рух або ефект створюється як окрема Action, після чого додається до NLA-системи з можливістю керування тривалістю, повторюваністю та точкою запуску. Це значно спрощує експорт анімацій у формат glTF, оскільки Three.js коректно імпортує всі анімаційні кліпи як незалежні сутності. У веб-додатку вони стають доступними через AnimationMixer, що дозволяє запускати, зупиняти або комбінувати анімації залежно від подій, не втручаючись у внутрішню структуру моделі.

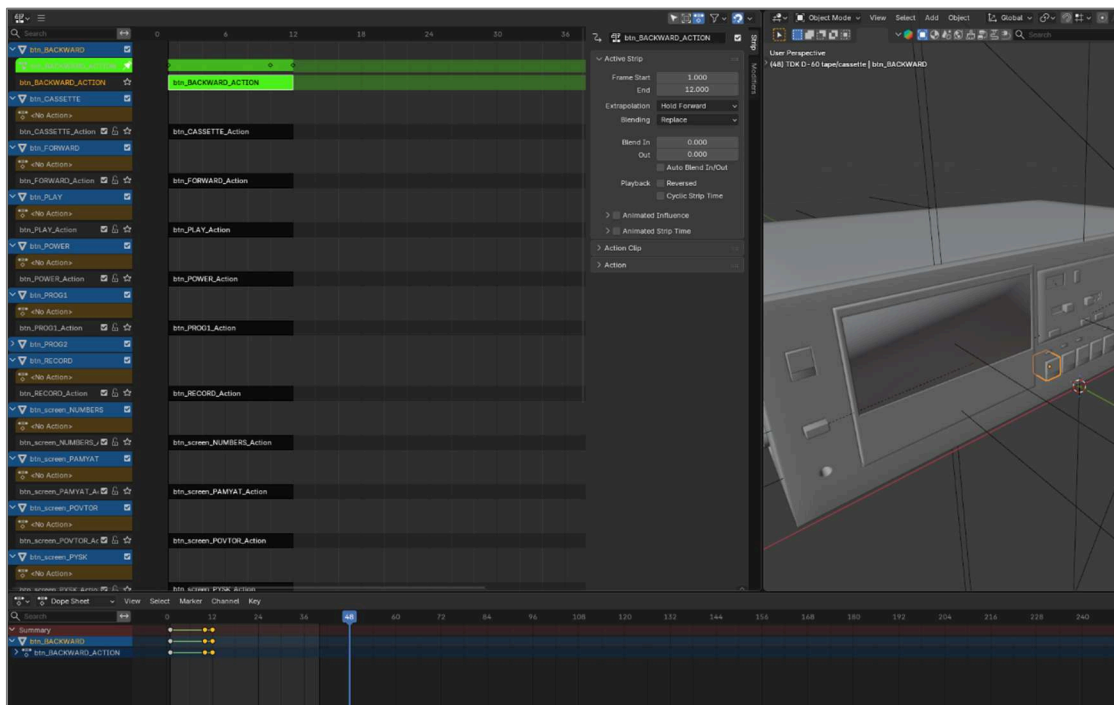


Рис. 3.3. Система NLA треків для анімації рухомих частин

З точки зору інтеграції з Three.js, підхід із NLA-треками є принципово важливим. Веб-додаток працює за подієвою логікою, де кожна дія користувача має викликати конкретну реакцію моделі. Короткі, логічно завершені анімаційні кліпи, створені через NLA, легко прив'язуються до таких подій. Наприклад, натискання кнопки може одночасно запускати кілька анімацій: механічний рух кнопки, зміну стану індикатора та початок обертання катушок. Усе це відбувається без необхідності

складного керування таймлайном у кодї, що зменшує кількість помилок і підвищує читабельність програмної логіки.

Окрему увагу було приділено тривалості анімаційних треків. Анімації, які логічно пов'язані між собою, створювалися з узгодженим таймінгом безпосередньо в Blender. Це дозволяє запускати їх у Three.js одночасно або послідовно без додаткового коригування швидкості чи затримок на рівні JavaScript. Такий підхід суттєво спрощує синхронізацію складних сценаріїв і робить поведінку моделі більш передбачуваною та стабільною.

Важливим аспектом анімацій стала робота з текстурами та матеріалами світлових елементів. Для кожного LED-елемента було створено окремий матеріал із власними параметрами емісії. Це дозволяє керувати світінням конкретного елемента, не впливаючи на інші частини моделі. У контексті Three.js та WebGL такий підхід є більш ефективним, ніж анімація одного спільного матеріалу, оскільки дає змогу точково змінювати emissiveIntensity або колір світіння лише там, де це необхідно, зменшуючи складність логіки та навантаження на систему.

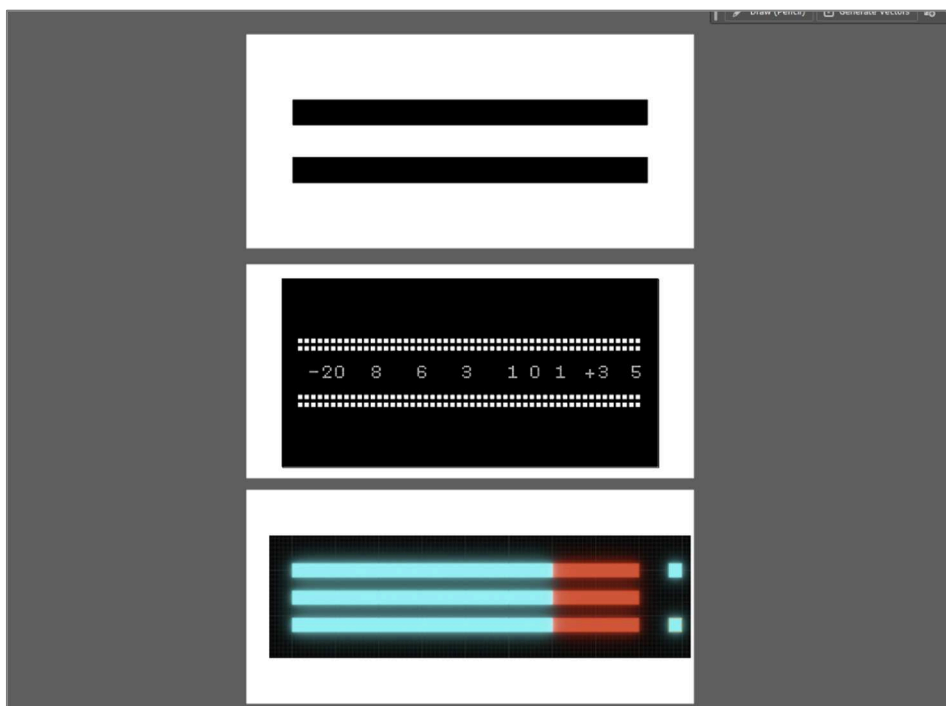


Рис. 3.4. Система масок для анімації VFD-індикатора

Найбільш складним з погляду анімації виявився VFD-індикатор. Пряма реалізація у вигляді окремої анімації для кожного світлового сегмента вимагала б керування приблизно сотнею незалежних «діодів», що є нераціональним для веб-середовища. Тому було застосовано багаторівневу систему масок. На нижньому рівні використовується базова текстурована карта екрана з двома лініями світіння, на середньому рівні – напівпрозора сітка, яка імітує структуру з приблизно 100 діодів, а на верхньому – дві рухомі смужки-маски. Довжина цих масок змінюється залежно від рівня аудіосигналу, що створює ілюзію роботи повноцінного VFD-індикатора.

Застосування масок замість індивідуальної анімації кожного сегмента дозволило значно зменшити кількість анімаційних елементів і підвищити продуктивність. При цьому візуальний результат залишається переконливим і близьким до поведінки реального пристрою. У поєднанні з NLA-треками та подієвою логікою Three.js такий підхід забезпечує гнучку, масштабовану та технічно обґрунтовану систему анімації, придатну для складних інтерактивних веб-експонатів.

Для оптимальної роботи моделі у веб-додатку її довелося серйозно оптимізувати. Для цього використали запікання деталей (Baking), яке дозволяє зберегти зовнішній вигляд моделі, але суттєво зменшити кількість полігонів. Спочатку була створена детальна High-poly версія з усіма фасками, дрібними виступами та гвинтами. Потім ця геометрія була “спроектована” на Low-poly модель за допомогою Normal Map. По суті, об’ємні елементи з High-poly перетворилися на інформацію про освітлення на текстурі. У результаті модель візуально залишилася такою ж деталізованою, але її сітка зменшилася до 30–40 тисяч трикутників (оптимальне значення для інтеграції у веб).

Після завершення оптимізації готуємо модель до інтеграції у Three.js. Для цього була виконана експортація у формат .glb – бінарну версію glTF 2.0, яка є найзручнішою для веб-графіки. Експортуємо зі стисненням Draco, яке суттєво зменшує вагу файлу. У нашому випадку розмір моделі скоротився до приблизно 6 МБ – це дозволяє завантажувати її за 2–3 секунди навіть через мобільний інтернет. Швидкість завантаження напряму впливає на перше враження користувача та загальну плавність роботи платформи.

Під час експорту були активовані наступні параметри: Apply Modifiers, щоб усі геометричні зміни стали частиною фінальної моделі, та Export Materials, щоб зберегти налаштування PBR-матеріалів. Завдяки цим налаштуванням модель виглядатиме у браузері саме так, як виглядала у Blender, без необхідності додаткових налаштувань з боку Three.js.

3.1.4. Завантаження GLB-моделі у Three.js та ініціалізація сцени

Інтеграція моделі у веб-середовище здійснювалася за допомогою бібліотеки Three.js, яка є одним із найзручніших інструментів для роботи з WebGL. На першому етапі була ініціалізована базова сцена: створено рендерер (WebGLRenderer) із ввімкненим згладжуванням (*antialias*), щоб краї моделі не виглядали “зубчастими”, та застосовано ACESFilmicToneMapping.

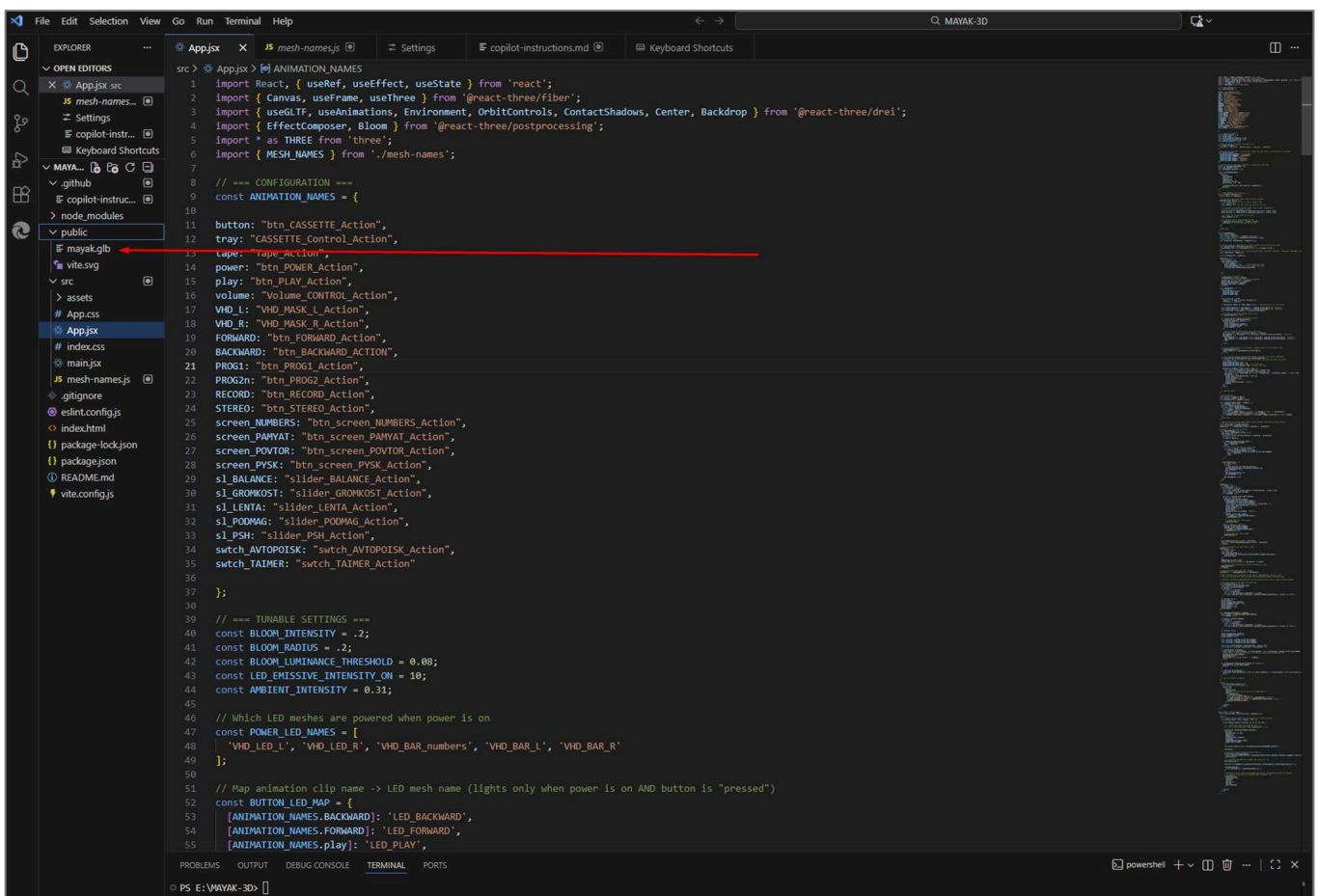


Рис. 3.5. Three.js з підключеною 3D моделлю

Цей режим тонального відображення дозволяє отримати більш природне й, фактично, “кінематографічне” освітлення – воно краще передає блиски, тіні та загальний контраст.

Далі у сцену завантажувався сам 3D-ассет за допомогою GLTFLoader. При завантаженні спрацьовує callback-функція, у якій модель додається до сцени, а скрипт виконує обхід усіх дочірніх об’єктів (traversal). Це необхідно, щоб правильно налаштувати поведінку кожного елемента в умовах реального часу. Зокрема, окремо активуються параметри тіней:

- `object.castShadow = true` – об’єкт може відкидати тінь,
- `object.receiveShadow = true` – об’єкт може приймати тінь від інших.

У контексті нашого експоната це критично для передачі реалістичної глибини та взаємодії між деталями. Без коректної роботи з тінями навіть високоякісна модель виглядатиме пласкою.

```
loader.load('mayak140.glb', (glTF) => {
  const model = glTF.scene;
  model.traverse((child) => {
    if (child.isMesh) {
      child.castShadow = true; // Об'єкт відкидає тінь
      child.receiveShadow = true; // Об'єкт приймає тінь
    }
  });
  scene.add(model);
});
```

3.1.5. Підготовка структури вузлів до інтерактивності

Щоб керувати моделлю з боку коду, спочатку потрібно отримати доступ до її інтерактивних елементів. Після того як сцена завантажується, виконується пошук об’єктів за їхніми іменами – саме тими, які були задані під час моделювання у Blender. Знайдені елементи зберігаються у глобальні змінні, щоб їх можна було вільно

використовувати у циклі анімації та в логіці взаємодії. Такий підхід дозволяє зручно контролювати кожну частину механіки: від обертання катушок до руху кнопок.

Окремо була реалізована робота з VFD-індикатором, оскільки він має складнішу структуру. Індикатор складається із серії дрібних мешів – сегментів, кожен з яких відповідає за окрему ділянку світіння. Усі ці сегменти зібрані в масив, завдяки чому до них можна звертатися через індекси. Це відкриває можливість програмно регулювати інтенсивність емісії (*emissiveIntensity*) кожного сегмента. Таким чином індикатор «реагує» на аудіосигнал, і його яскравість змінюється відповідно до рівня звуку, що створює ефект справжнього електронного дисплея.

3.1.6. Тестування коректності геометрії, pivot-точок і анімацій після експорту

Фінальним етапом імплементації стала перевірка моделі вже безпосередньо у браузері. За допомогою інструментів розробника та візуальних хелперів, таких як AxesHelper, була протестована правильність розташування осей обертання для катушок і регуляторів.



Рис. 3.6. Запуск і тестування інтерактивної 3D моделі на локальному сервері

Під час тестів також перевірялася плавність обертання та відсутність артефактів шейдингу. Результати показали, що анімація працює стабільно, без ривків і викривлень, а матеріали – металеві й пластикові поверхні – адекватно реагують на освітлення сцени. Віртуальні джерела світла правильно підсвічують фаски, створюють м'які тіні та підсилюють відчуття глибини моделі.

У сукупності ці перевірки підтвердили, що 3D-ассет технічно готовий до подальшого етапу – впровадження інтерактивної логіки та поведінки елементів у рамках платформи.

3.2. Створення інтерактивної логіки та аудіовізуальних модулів

Після успішного імпорту 3D-моделі у сцену розпочався етап програмування логіки роботи інтерактивного експоната. Модель, яка доти була звичайним статичним об'єктом, потрібно було перетворити на повноцінний веб-додаток, здатний реагувати на дії користувача та обробляти дані в реальному часі. Іншими словами, йшлося не лише про візуальну частину, а про створення системи поведінки, яка робить експонат «живим» у браузері.

Архітектура взаємодії була побудована на подієвій моделі JavaScript. Це дало змогу організувати керування через набір подій – натискання кнопок, переміщення елементів, запуск і зупинку анімацій. Подієвий підхід дозволяє відокремити логіку інтерфейсу від логіки моделі, що робить структуру коду більш зрозумілою, а сам застосунок – гнучким і легшим у підтримці.

3.2.1. Програмування взаємодії з моделлю у Three.js

Оскільки браузер фактично відображає 3D-сцену як звичайний набір пікселів у межах елемента <canvas>, виникає проблема: він «не знає» про існування об'єктів у тривимірному просторі. Тому для реалізації кліків по кнопках і регуляторах використовується механізм Raycasting – по суті, віртуальне «кидання променя», який визначає, у що саме натиснув користувач.

Алгоритм взаємодії працює так:

1. Під час кліку миші зчитуються координати курсору, після чого вони нормалізуються у діапазон від -1 до $+1$ – у координатну систему, з якою працює WebGL.

2. Об'єкт THREE.Raycaster випускає невидимий промінь із позиції камери через точку, де знаходиться курсор користувача.

3. Промінь перевіряє перетини з масивом інтерактивних об'єктів – кнопками, регуляторами та іншими елементами, які були кешовані під час завантаження моделі.

Якщо Raycaster знаходить збіг, скрипт зчитує ім'я об'єкта (наприклад, Button_Play) і викликає відповідну функцію контролера, яка відповідає за логіку взаємодії.

Щоб користувач отримувач візуальне підтвердження натискання, для кнопок додано невелику процедурну анімацію: елемент плавно зміщується по локальній осі Z, імітуючи хід реального механізму, а після цього повертається у вихідну позицію. Такий підхід робить взаємодію приємнішою й більш натуралістичною.

3.2.2. Інтеграція Web Audio API для керування анімаціями та індикаторами

Для того щоб візуальна частина експоната працювала синхронно зі звуком, був створений окремий модуль аудіосистеми на основі Web Audio API. Під час запуску додатку ініціалізується AudioContext, усередині якого формується повний граф обробки сигналу – від джерела до фінального виходу на динаміки. Це дозволяє гнучко керувати звуком та отримувати дані для анімації у режимі реального часу.

Ланцюг аудіових вузлів побудований у такій послідовності: MediaElementSource (джерело: Web Audio API) → AnalyserNode (аналіз спектра та амплітуди) → GainNode (керування гучністю) → AudioContext.destination (фізичний вихід на динаміки).

Основним елементом цього графа є AnalyserNode. Він працює з параметром `fftSize = 256`, що забезпечує хороший баланс між точністю аналізу та швидкістю

обробки даних. Важливо, що цей вузол не змінює сам звук – усі аудіоефекти проходять повз нього – але він у будь-який момент може надати два типи масивів:

- Time Domain Data – амплітуда сигналу у часовій області,
- Frequency Data – розподіл енергії по частотах.

Ці дані використовуються для керування візуальними елементами моделі: наприклад, зміни яскравості сегментів VFD-індикатора або плавності обертання механічних елементів. Завдяки цьому експонат реагує на звук у реальному часі, створюючи ефект роботи справжнього пристрою.

3.2.3. Реалізація динамічного VFD-індикатора та анімації механіки

Візуалізація роботи магнітофона реалізується через головний цикл рендерингу `requestAnimationFrame`, який оновлюється приблизно 60 разів на секунду. Саме цей цикл відповідає за плавність анімацій і синхронність між звуком, механікою та світловими ефектами.

Анімація завантаження носія. Для підвищення імерсивності реалізовано сценарій інтерактивного вставляння касети. За замовчуванням касетоприймач порожній. При натисканні кнопки Play (або окремої зони лотка) запускається послідовність анімацій, керована бібліотекою GSAP:

1. 3D-модель касети переміщується з позиції «поза кадром» у точку перед лотком.
2. Касета плавно засувається всередину механізму (зміна координати Z).
3. Кришка касетоприймача закривається (обертання по осі X).

Лише після завершення цієї послідовності (`onComplete`) активується логіка відтворення звуку та обертання вузлів. Це створює відчуття фізичної взаємодії з пристроєм, де підготовка до роботи займає певний час, як у реальному житті.

Анімація VFD-індикатора. Вакуумно-люмінесцентний індикатор у моделі складається з окремих сегментів-мешів, зібраних у масив. На кожному кадрі скрипт отримує з `AnalyserNode` масив даних, обчислює середньоквадратичне значення гучності (RMS) та нормалізує його до робочого діапазону. На основі цього значення відбувається зміна параметра `emissiveIntensity` – сили світіння сегментів індикатора.

Щоб індикатор виглядав реалістично, додано алгоритм згладжування. Він імітує інерційність фізичного VFD: стовпчики не зникають миттєво при падінні рівня сигналу, а поступово згасають. Завдяки цьому цифрова модель поводить себе подібно до справжнього приладу, де світіння має невеликий «хвіст» після кожного піку.

Анімація механіки. Рух катушок синхронізований зі станом аудіоплеєра. Якщо плеєр у режимі відтворення (`isPlaying = true`), обидві катушки починають обертатися навколо своїх осей. Швидкість обертання кожної з них динамічно змінюється залежно від умовного діаметра намотаної стрічки. Це дозволяє імітувати природну поведінку касетного механізму: приймальна катушка поступово пришвидшується, тоді як подавальна – сповільнюється. Такий підхід робить анімацію не просто декоративною, а фізично обґрунтованою.

3.3.Тестування та оцінка ефективності інтерактивної експозиції

Фінальним етапом практичної розробки стала перевірка того, наскільки зручно та ефективно працює створена веб-платформа. Для музейного експоната важливо не лише уникнути технічних помилок, а й зробити взаємодію інтуїтивною та навчально корисною. У цьому підрозділі наведено результати тестування на різних пристроях і спостереження за тим, як користувачі взаємодіяли з цифровим двійником магнітофона «Маяк-140». Для реалізації інтерактивних функцій (обертання, демонстрація роботи вузлів) було застосовано принципи комп'ютерної анімації, включаючи роботу з ключовими кадрами та застосуванням фізичних законів руху, як це висвітлено у практикумі, підготовленому кафедрою комп'ютерних мультимедійних технологій ДУ «КАІ» (НАУ) [48].

3.3.1.Юзабіліті-тестування інтерфейсу та взаємодій

Перевірка зручності проходила у форматі сценарного тестування. Учасникам пропонувалося виконати базову послідовність дій: «Увімкнути живлення → Знайти трек → Запустити відтворення → Налаштувати еквалайзер».

Під час тестів виявлено кілька типових проблем для 3D-інтерфейсів:

– «Втрата фокусу» камери. Користувачі інколи надто далеко віддаляли камеру або навпаки – випадково «потрапляли» всередину моделі. Рішення: у параметрах *OrbitControls* були обмежені кути обертання (*maxPolarAngle*) та допустима дистанція зуму (*minDistance*, *maxDistance*). Камера тепер рухається лише в межах, які гарантують комфортний огляд експоната.

– Низька точність натискань на мобільних пристроях. Дрібні кнопки виявилися складними для взаємодії пальцем. Рішення: навколо таких елементів додано невидимі «хітбокси» – прозорі геометричні примітиви більшого розміру. Вони перехоплюють події *Raycast*, не змінюючи зовнішній вигляд моделі.

У результаті гібридна схема інтерфейсу – комбінація 3D-сцени та 2D-оверлею – продемонструвала хорошу ефективність: користувачі швидко освоювали керування й інтуїтивно розуміли, де знаходяться функції пошуку, а де – елементи механіки.

3.3.2. Оцінка освітньої ефективності цифрового експоната

Основна ідея розробки полягала в тому, що інтерактивна візуалізація здатна покращити розуміння технічних процесів, які зазвичай залишаються невидимими. Для перевірки цього аналізувалася взаємодія користувачів із еквалайзером та VFD-індикатором.

Результати підтвердили, що аудіо-візуальний зв'язок працює ефективно. Користувачі швидко помічали залежність між зміною положення повзунків еквалайзера та реакцією індикатора. Особливо показово виявилася можливість експериментувати з «перевантаженням» сигналу: підняття рівня до +5 dB і подальше прослуховування спотворень допомогли пояснити поняття динамічного діапазону на практиці, а не лише на словах.

У порівнянні зі статичними схемами та текстовими поясненнями, візуалізація спектра через VFD виявилась значно ефективнішою. Динаміка стовпчиків буквально дозволила користувачам «побачити звук», що зробило абстрактні поняття амплітуди й частоти набагато зрозумілішими.

3.3.3. Аналіз результатів та підтвердження практичної цінності вебплатформи

Перевірка технічної частини показала високу стабільність роботи прототипу на різних пристроях:

- Оптимізація: завдяки GLB-формату та Draco-стисненню розмір моделі не перевищує ~15 МБ. Це забезпечує завантаження сторінки за менш ніж 2 секунди (First Contentful Paint < 2 с).
- Продуктивність: на ПК система стабільно працює на 60 FPS; на смартфонах середнього класу – 30–45 FPS, що є комфортним для взаємодії.
- Синхронізація: затримка між звуком і реакцією VFD-індикатора мінімальна, а отже, користувач сприймає модель як цілісний фізичний пристрій.
- З погляду практичної цінності, створений веб-експонат є повністю готовим для використання у музейних або навчальних середовищах. Архітектура проєкту дозволяє легко додавати нові моделі техніки без змін у базовому коді – наприклад, інші магнітофони серій «Юпітер» чи «Олімп». Це робить платформу універсальним інструментом для збереження технічної спадщини та популяризації історії аудіотехнологій України в цифровому форматі.

ВИСНОВКИ

У ході виконання кваліфікаційної роботи комплексно досліджена та розв'язана задача зі створення веб-платформи для інтерактивної демонстрації технічних пристроїв, взявши за основу унікальний магнітофон «Маяк-140». Результати дослідження підсумовують наступне:

1. Теоретичний базис. Аналіз принципів мультимедійного навчання (зокрема робіт Маєра та Свеллера) підтвердив, що пояснення невидимих процесів – таких як електромагнітна індукція чи частотний спектр – потребує макроскопічної візуалізації. Саме тому було запропоновано концепцію аудіо-візуальної синергії. Суть її проста: механіка та індикатори на екрані рухаються синхронно зі звуком, даючи користувачеві можливість інтуїтивно зрозуміти фізику процесу, не заглиблюючись у складні формули.

2. Технологічний фундамент. Вибір інструментів був продиктований пошуком балансу між картинкою та продуктивністю. Зв'язка Blender (для моделювання та запікання текстур), Three.js (для рендерингу) та Web Audio API (для роботи зі звуком) показала себе найкраще. Цей стек дозволив отримати якісну візуалізацію (PBR-матеріали) і при цьому залишити платформу доступною для мобільних пристроїв, уникнувши проблем, властивих «важким» ігровим рушіям.

3. Методика моделювання. У ході роботи була відпрацьована методика створення цифрового двійника для вебу. Вона базується на декомпозиції об'єкта: кожен інтерактивний вузол моделюється окремо. Використання карт нормалей дозволило суттєво оптимізувати геометрію, а правильне налаштування ієрархії сцени дало змогу програмно керувати анімацією без зайвих складнощів.

4. Програмна реалізація. Результатом стала працююча веб-платформа на архітектурі SPA. Головна «фішка» – це механізм керування VFD-індикатором. Дані про амплітуду сигналу беруться прямо з аудіопотоку і «на льоту» перетворюються на анімацію 3D-сегментів. Це забезпечило повну синхронізацію того, що користувач чує, з тим, що він бачить.

5. Практична цінність. Тести показали, що платформа працює стабільно (60 FPS на ПК, понад 30 FPS на смартфонах), а гібридний інтерфейс є зручним для користувача. Створений експонат повністю готовий до використання в цифрових музеях та освітніх проєктах. Він слугує інструментом збереження технічної спадщини (Digital Heritage) і допомагає популяризувати інженерні знання.

Можна стверджувати, що мета роботи досягнута: складний технічний артефакт перетворився на зрозумілий, доступний і, що важливо, цікавий об'єкт для дослідження в цифровому просторі.

СПИСОК БІБЛІОГРАФІЧНИХ ПОСИЛАНЬ ВИКОРИСТАНИХ ДЖЕРЕЛ

1. Art Term: Multi-media [Електронний ресурс] // Tate Gallery. – URL: <https://www.tate.org.uk/art/art-terms/m/multi-media> (дата звернення: 10.11.2025).
2. Гатауліна А. Інтерактивне та мультимедійне обладнання у школі // Відкритий урок: розробки, технології, досвід. – 2009. – № 2. – С. 27–29.
3. Edwards B. Forty Years Ago, the Mac Triggered a Revolution in User Experience [Електронний ресурс] // Smithsonian Magazine. – 2024. – URL: <https://www.smithsonianmag.com/innovation/forty-years-ago-the-mac-triggered-a-revolution-in-user-experience-180983623> (дата звернення: 10.11.2025).
4. Macromedia Director [Електронний ресурс] // Macromedia Fandom. – URL: https://macromedia.fandom.com/wiki/Macromedia_Director (дата звернення: 10.11.2025).
5. Rhodes M. How Adobe Flash, once the face of the web, fell to the brink of obscurity—and why it’s worth saving [Електронний ресурс] // Quartz. – 2017. – URL: <https://qz.com/863467/how-adobe-flash-once-the-face-of-the-web-fell-to-the-brink-of-obscurity-and-why-its-worth-saving> (дата звернення: 10.11.2025).
6. Burmistrov I. Touchscreen Kiosks in Museums: User Experience Analysis [Електронний ресурс] / I. Burmistrov // ResearchGate. – 2015. – URL: https://www.researchgate.net/publication/283905642_Touchscreen_Kiosks_in_Museums (дата звернення: 10.11.2025).
7. WebGL: 2D and 3D graphics for the web [Електронний ресурс] // MDN Web Docs. – URL: https://developer.mozilla.org/en-US/docs/Web/API/WebGL_API (дата звернення: 10.11.2025).
8. Taylor J. Talking About Multimedia: A Layered Design Framework // Journal of Educational Media. – 2000. – Vol. 25, № 2. – С. 135–146.

9. Mayer R. E., Moreno R. Nine ways to reduce cognitive load in multimedia learning // Educational Psychologist. – 2003. – Vol. 38, № 1. – P. 43–52. DOI: 10.1207/S15326985EP3801_6.
10. Mayer R. E. The Cambridge Handbook of Multimedia Learning. – Cambridge : Cambridge University Press, 2005. – 662 с.
11. Sweller J. Cognitive load during problem solving: Effects on learning // Cognitive Science. – 1988. – Vol. 12, № 2. – С. 257–285. DOI: 10.1207/s15516709cog1202_4.
12. Jonassen D. H. Learning to Solve Problems: An Instructional Design Guide. – San Francisco : Jossey-Bass, 2004. – 463 с.
13. Tufte E. R. Visual Explanations: Images and Quantities, Evidence and Narrative. – Cheshire, CT : Graphics Press, 1997. – 156 с.
14. Schnotz W., Lowe R. A unified view of learning from animated and static graphics // The Cambridge Handbook of Multimedia Learning / ed. by R. E. Mayer. – Cambridge : Cambridge University Press, 2005. – с. 549–568.
15. Ware C. Information Visualization: Perception for Design. – 3rd ed. – Burlington : Morgan Kaufmann, 2013. – 512 с.
16. Lowe R. Animation and learning: Value for money? // British Journal of Educational Technology. – 2004. – Vol. 35, № 4. – С. 423–436.
17. Kahneman D., Knetsch J. L., Thaler R. H. Anomalies: The Endowment Effect, Loss Aversion, and Status Quo Bias // Journal of Economic Perspectives. – 1991. – Vol. 5, № 1. – С. 193–206.
18. Loewenstein G. The Psychology of Curiosity: A Review and Reexamination // Psychological Bulletin. – 1994. – Vol. 116, № 1. – С. 75–98.
19. Хартія ЮНЕСКО про збереження цифрової спадщини [Електронний ресурс] / ЮНЕСКО ; прийнято 15 жовтня 2003 р. – Париж, 2003. – URL: <https://unesdoc.unesco.org/ark:/48223/pf0000179529> (дата звернення: 16.11.2025).
20. MicroPasts [Електронний ресурс] : спільний краудсорсинговий дослідницький проект / Британський музей ; Університетський коледж Лондона (UCL). – URL: <https://micropasts.org> (дата звернення: 16.11.2025).

21. Круг С. Не змушуйте мене думати: Розсудливий підхід до зручності в користуванні сайтами та мобільними застосунками / пер. з англ. Л. Горин. – Харків : ArtHuss, 2024. – 198 с.
22. Web Content Accessibility Guidelines (WCAG) [Електронний ресурс] / World Wide Web Consortium (W3C) ; Web Accessibility Initiative (WAI). – URL: <https://www.w3.org/WAI/standards-guidelines/wcag/> (дата звернення: 16.11.2025).
23. Віртуальні тури [Електронний ресурс] / Смітсонівський національний музей природознавства. – URL: <https://naturalhistory.si.edu/visit/virtual-tour> (дата звернення: 16.11.2025).
24. The British Museum [Електронний ресурс] : 3D models // Sketchfab. – URL: <https://sketchfab.com/britishmuseum> (дата звернення: 16.11.2025).
25. The Musical Instrument Museum (MIM) [Електронний ресурс] // Atlas Obscura. – URL: <https://www.atlasobscura.com/places/the-musical-instrument-museum-in-brussels-brussels-belgium> (дата звернення: 16.11.2025).
26. Nielsen J. Usability Engineering. – San Francisco : Morgan Kaufmann, 1993. – 362 р.
27. Garrett J. J. The Elements of User Experience: User-Centered Design for the Web and Beyond. – 2nd ed. – Berkeley : New Riders, 2011. – 192 с.
28. Web Content Accessibility Guidelines (WCAG) 2.1 [Електронний ресурс] / World Wide Web Consortium. – 2018. – URL: <https://www.w3.org/TR/WCAG21/> (дата звернення: 19.11.2025).
29. Maya vs Blender: Best 3D Software Comparison 2025 [Електронний ресурс] // Innovecs Games. – URL: <https://www.innovecsgames.com/blog/maya-vs-blender/> (дата звернення: 19.11.2025).
30. Blender vs Cinema 4D: The Ultimate Guide [Електронний ресурс] // School of Motion. – URL: <https://www.schoolofmotion.com/blog/blender-vs-cinema-4d> (дата звернення: 19.11.2025).
31. Why Blender Is Better Than Maya (and why you should switch) in 2024 [Електронний ресурс] // InspirationTuts : [YouTube-канал]. – 2023. – 22 червня. – URL: <https://www.youtube.com/watch?v=4r4ON1c6hPM> (дата звернення: 19.11.2025).

32. WebGL Specification [Електронний ресурс] / Khronos Group. – URL: <https://www.khronos.org/webgl/> (дата звернення: 19.11.2025).
33. What is an OBJ file? [Електронний ресурс] // Adobe. – URL: <https://www.adobe.com/creativecloud/file-types/image/vector/obj-file.html> (дата звернення: 19.11.2025).
34. glTF vs FBX: Which format should I use? [Електронний ресурс] // Threekit. – 2019. – 19 листопада. – URL: <https://www.threekit.com/blog/gltf-vs-fbx-which-format-should-i-use> (дата звернення: 19.11.2025).
35. McCurdy D. Three.js – GLTFLoader Documentation [Електронний ресурс]. – URL: <https://threejs.org/docs/#examples/en/loaders/GLTFLoader> (дата звернення: 19.11.2025).
36. Web Audio API [Електронний ресурс] / W3C. – URL: <https://www.w3.org/TR/webaudio/> (дата звернення: 19.11.2025).
37. Rendering on the Web: Single Page Applications [Електронний ресурс] // Google Developers. – URL: <https://developers.google.com/web/updates/2019/02/rendering-on-the-web> (дата звернення: 20.11.2025).
38. Three.js Performance Tips [Електронний ресурс] // DiscoverThree.js. – URL: <https://discoverthreejs.com/tips-and-tricks/performance/> (дата звернення: 20.11.2025).
39. Магнітофон-приставка «Маяк МП-140С» [Електронний ресурс] // REM-TV.NET. – URL: <https://rem-tv.net/uk/stuff/mayak-mp-140s> (дата звернення: 20.11.2025).
40. glTF 2.0 Specification: PBR Materials [Електронний ресурс] // Khronos Group. – URL: <https://www.khronos.org/gltf/> (дата звернення: 20.11.2025).
41. glTF Viewer (Khronos Group) [Електронний ресурс] / Khronos Group. – URL: <https://github.khronos.org/gltf-Sample-Viewer-Release/> (дата звернення: 20.11.2025).
42. Consistency in UI Design [Електронний ресурс] // Interaction Design Foundation. – URL: <https://www.interaction-design.org/literature/topics/consistency> (дата звернення: 20.11.2025).

43. Google Material Design – Responsive Layout Grid [Електронний ресурс] // Google. – URL: <https://m3.material.io/foundations/adaptive-design/overview> (дата звернення: 20.11.2025).

44. Nielsen J. 10 Usability Heuristics for User Interface Design [Електронний ресурс] // Nielsen Norman Group. – URL: <https://www.nngroup.com/articles/ten-usability-heuristics/> (дата звернення: 20.11.2025).

45. Мультимедійні технології в освіті та інших сферах діяльності : матеріали XV Міжнар. наук.-практ. конф. (Київ, 6 листоп. 2024 р.). Київ : ДУ «КАІ» (НАУ), 2024. 376 с.

46. Денисенко С. М. Основи композиції і проєктної графіки : навч. посіб. [Електронне видання]. Київ : НАУ, 2021. 52 с.

47. Ковальов Ю. М. Основи геометричного моделювання : навч. посіб. Київ : Вища шк., 2003. 231 с.

48. Основи комп'ютерної анімації : лабораторний практикум / уклад.: О. А. Бобарчук, О. Г. Батрак, І. А. Гніденко. Київ : НАУ, 2024. 58 с.

ДОДАТКИ

Додаток А

Карта текстур Маяк-140С

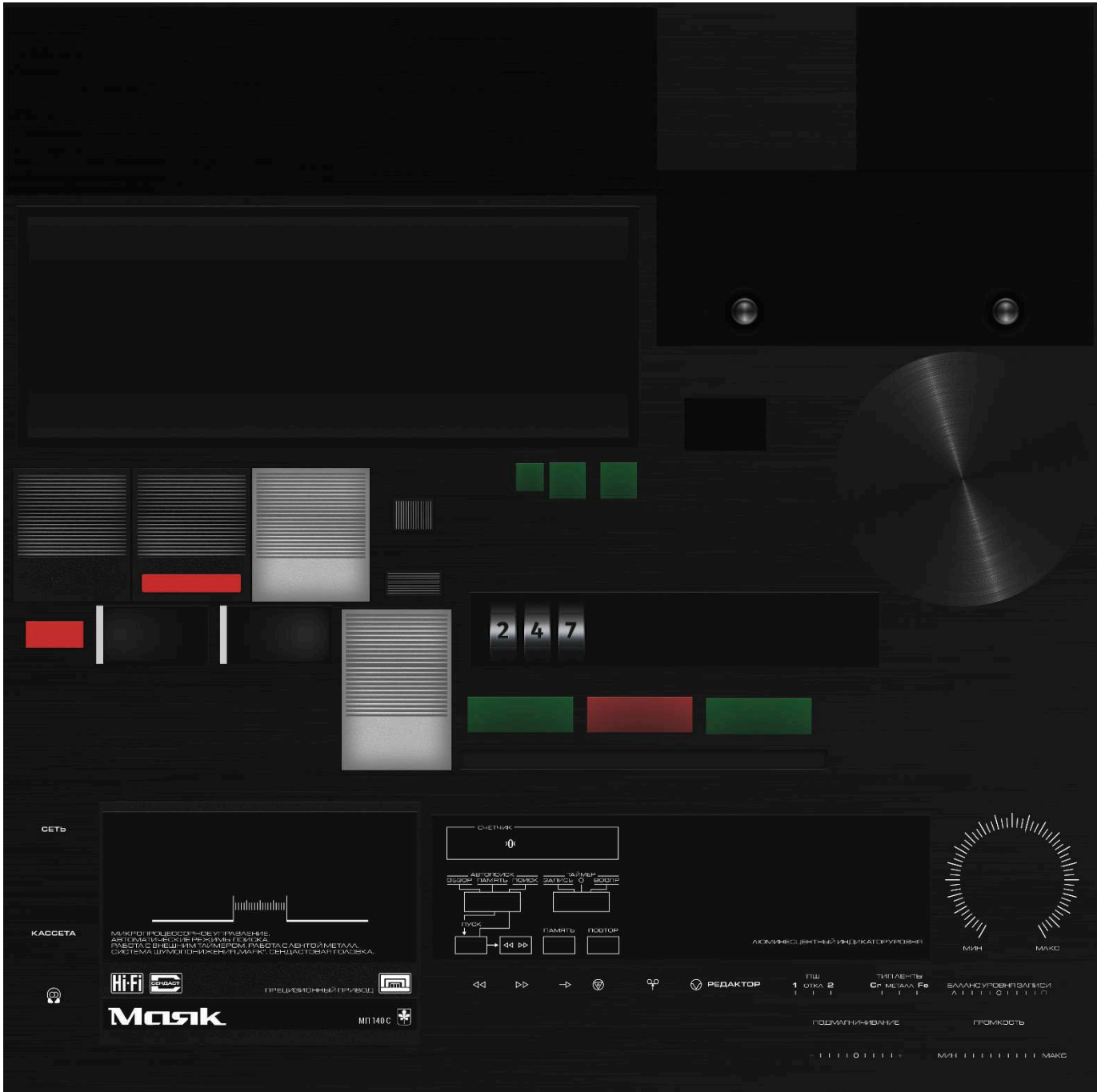


Рис. А.1. Карта текстур Маяк-140С