МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ  
НАЦІОНАЛЬНИЙ ТЕХНІЧНИЙ УНІВЕРСИТЕТ УКРАЇНИ  
«КИЇВСЬКИЙ ПОЛІТЕХНІЧНИЙ ІНСТИТУТ  
імені ІГОРЯ СІКОРСЬКОГО»

НАВЧАЛЬНО-НАУКОВИЙ ІНСТИТУТ АТОМНОЇ ТА ТЕПЛОВОЇ ЕНЕРГЕТИКИ

КАФЕДРА ЦИФРОВИХ ТЕХНОЛОГІЙ В ЕНЕРГЕТИЦІ

**ВІЗУАЛІЗАЦІЯ ГРАФІЧНОЇ ТА ГЕОМЕТРИЧНОЇ ІНФОРМАЦІЇ**

РОЗРАХУНКОВО-ГРАФІЧНА РОБОТА

“Spatial audio”

Виконав:

студент групи ТР-32мп

Мар’яш Дмитро Ігорович

Київ – 2024

1. **Завдання**

* Імплементувати просторове аудіо за допомогою WebAudio HTML5 API, використовуючи код з практичного завдання 2.
* Імплементувати обертання джерела звуку відносно геометричного центру поверхні з використанням дотичного інтерфейсу (смартфону, що використовує сенсори для обробки інформації, щодо положення в просторі). На відміну від попередніх робіт поверхня залишається нерухомою - рухається джерело звуку. Відтворити улюблену пісню в форматі mp3/ogg, з можливістю контролювати місцезнаходження джерела звуку в просторі користувачем.
* Візуалізувати джерело звуку за допомогою сфери.
* Додати звуковий фільтр за варіантом (використовуючи інтерфейс BiquadFilterNode). Додати інтерфейс користувача, який би вмикав/вимикав фільтр. Встановити параметри фільтру відповідно до вподобань.
* Підготувати звіт в цифровому вигляді, який би містив необхідні частини, що сповна описують поставлені задачі та виконану роботу.

1. **Теоретичні відомості**

**WebAudio API**

Раніше аудіо в Інтернеті було досить примітивним і вимагало використання плагінів, таких як Flash і QuickTime. Введення аудіоелемента в HTML5 стало значним кроком вперед, дозволяючи відтворювати базове потокове аудіо без сторонніх розширень. Однак для більш складних аудіозадач, таких як інтерактивні програми або сучасні веб-ігри, необхідне більш потужне рішення. Саме для цього була розроблена специфікація WebAudio API, яка включає можливості сучасних ігрових звукових рушіїв, а також завдання мікшування, обробки та фільтрації, що є в сучасних настільних програмах для створення звуку.

WebAudio API розроблено з урахуванням широкого спектру сценаріїв використання, включаючи можливість підтримки будь-яких варіантів, які можна реалізувати за допомогою оптимізованого механізму C++ з керуванням через сценарій, що запускається в браузері. Хоча сучасне настільне аудіопрограмне забезпечення, таке як Apple Logic Audio, може мати дуже розширені можливості, деякі з яких важко або неможливо відтворити у веб-середовищі, WebAudio API все ж здатен підтримувати широкий спектр складних ігор та інтерактивних програм, включаючи музичні додатки. Цей API також може бути чудовим доповненням до розширених графічних функцій, які пропонує WebGL. У перспективі можливе додавання ще більш розширених можливостей.

WebAudio API дозволяє створювати, маніпулювати і контролювати аудіо в реальному часі в веб-браузері. Він забезпечує низку функцій, таких як:

* Обробка сигналів: Можливість створення та обробки звукових сигналів за допомогою різноманітних ефектів.
* Мікшування звуку: Об'єднання декількох аудіопотоків з можливістю зміни гучності, панорамування та інших параметрів.
* Просторове аудіо: Моделювання звуку в тривимірному просторі, що дозволяє створювати ефекти позиціювання звуку.
* Аналіз звуку: Отримання даних про звуковий сигнал для створення візуалізацій або обробки звуку в реальному часі.

**BiquadFilterNode**

BiquadFilterNode – це процесор AudioNode, що реалізує популярні фільтри нижчого порядку, які є будівельними блоками для базових регуляторів тембру (баси, середні та високі частоти), графічних еквалайзерів та інших складних фільтрів. Кілька BiquadFilterNode можуть бути об'єднані для створення більш складних фільтрів. Параметри фільтра, такі як частота, можна змінювати з часом для динамічного налаштування фільтра.

Кожен BiquadFilterNode можна налаштувати на один із кількох загальних типів фільтрів:

* Низькочастотний фільтр (Low-pass filter): Пропускає частоти нижче граничної частоти і зменшує частоти вище неї. Реалізує стандартний резонансний низькочастотний фільтр другого порядку зі спадом 12 дБ/октаву.
* Високочастотний фільтр (High-pass filter): Пропускає частоти вище граничної частоти і зменшує частоти нижче неї. Реалізує стандартний резонансний високочастотний фільтр другого порядку зі спадом 12 дБ/октаву.
* Смуговий фільтр (Band-pass filter): Пропускає діапазон частот і зменшує частоти нижче та вище цього діапазону. Реалізує смуговий фільтр другого порядку.
* Фільтр Lowshelf: Пропускає всі частоти, але додає підсилення або ослаблення на нижніх частотах. Реалізує фільтр низького рівня другого порядку.
* Фільтр Highshelf: Пропускає всі частоти, але додає підсилення на вищих частотах. Реалізує фільтр високої полиці другого порядку.
* Піковий фільтр (Peaking filter): Пропускає всі частоти, але додає підсилення або ослаблення на певному діапазоні частот.
* Режекторний фільтр (Notch filter): Пропускає всі частоти, крім певного діапазону частот. Це протилежність смуговому фільтру.
* Теоретичні відомості про фільтри
* Фільтри є ключовими елементами в аудіообробці, і вони класифікуються за їхніми частотними характеристиками:
* Низькочастотні фільтри (Low-pass filters) дозволяють низьким частотам проходити без змін, але послаблюють високі частоти. Вони використовуються для усунення високочастотних шумів або для створення тепліших звуків.
* Високочастотні фільтри (High-pass filters) пропускають високі частоти і послаблюють низькі. Вони корисні для видалення низькочастотних шумів або для надання звуку більш яскравого характеру.
* Смугові фільтри (Band-pass filters) пропускають лише певний діапазон частот, послаблюючи всі інші. Вони використовуються для ізоляції певних частот з аудіосигналу.
* Режекторні фільтри (Notch filters) роблять протилежне смуговим фільтрам, видаляючи вузький діапазон частот і пропускаючи всі інші. Вони часто використовуються для видалення специфічних частот, наприклад, 50/60 Гц гулу електричних мереж.

1. **Аспекти імплементації**

Було реалізовано обертання джерела звуку навколо геометричного центру ділянки поверхні круговим способом протягом певного часу. Цього разу поверхня залишається нерухомою, а джерело звуку переміщується, що додає до процесу реалізму і дозволяє користувачеві відчути ефект руху звуку в просторі. Джерело звуку візуалізовано у вигляді сферичної геометрії, що покращує розуміння його положення у тривимірному просторі.

A screenshot of a computer

Description automatically generated

Рисунок 3.1 – Демонстрація практичного завдання №2

Цей функціонал у майбутньому дозволить ефективно керувати положенням джерела звуку у просторі всередині програми. Перший крок до реалізації просторового аудіо полягав у створенні HTML-елемента <audio>, який містить інформацію про аудіо-доріжку, в даному випадку це "caya" у форматі mp3. Цей елемент керування дозволяє зупиняти та продовжувати відтворення аудіо-доріжки, перемотувати її на потрібний час, керувати гучністю, а також надає можливість інтеграції додаткових функцій через JavaScript.

Наступним кроком було використання JavaScript та WebAudio API для створення об'єкта аудіоконтексту (AudioContext). Для цього контексту було створено та під'єднано три основні об'єкти:

1. Джерело звуку (MediaElementSource) – це об'єкт, який дозволяє нам використовувати HTML-елемент <audio> як джерело звуку для аудіоконтексту.
2. Об'єкт обробки просторового аудіо (Panner) – цей об'єкт дозволяє позиціонувати звук у тривимірному просторі, змінюючи його положення відносно слухача.
3. Звуковий фільтр (BiquadFilter) – об'єкт, що дозволяє застосовувати різні типи звукових фільтрів для обробки аудіосигналу.

Згідно з варіантом було обрано “Шелфовий фільтр високих частот”. Цей тип фільтра дозволяє змінювати рівень високих частот у сигналу. Параметри цього фільтра налаштовуються таким чином:

* Frequency (частота) – визначає нижню межу частот, на які застосовується підсилення або ослаблення.
* Q (ширина смуги) – не використовується в цьому типі фільтра.
* Gain (підсилення) – визначає рівень підсилення (у дБ), яке буде застосовано до частот вище встановленої частоти. Якщо значення від'ємне, частоти послаблюються.

Джерело звуку, обробка якого в просторі здійснюється об'єктом класу Panner, візуалізовано у WebGL-контексті у вигляді сфери. Це дозволяє переміщувати джерело звуку, змінюючи параметри об'єкта Panner, щоб створити ефект просторового переміщення звуку відповідно до місцезнаходження сфери у системі координат. Для налаштування стерео зображення використовуються чотири слайдери, які дозволяють регулювати такі параметри:

* Convergence (збіжність) – налаштовує точку сходження ліній зору, що впливає на глибину стерео ефекту.
* Eye separation (відстань між очима) – визначає відстань між "очима" віртуальної камери, що впливає на сприйняття тривимірності.
* Field of view (поле зору) – визначає кут огляду камери.
* Near clipping distance (відстань ближньої площини, що відсікає зображену на екрані геометрію) – визначає мінімальну відстань до об'єктів, які будуть відображені на екрані.

A person wearing sunglasses and a white shirt

Description automatically generated

Рисунок 3.2 – Демонстрація розрахунково-графічної роботи

1. **Інструкція користувача**

Для налаштування стерео зображення використовуються 4 слайдери, в який можна налаштувати наступні параметри:

* Convergence (збіжність)
* Eye separation (відстань між очима)
* Field of view (поле зору)
* Near clipping distance (відстань ближньої площини, що відсікає зображену на екрані геометрію)



Рисунок 4.1 – Контролер слайдерів параметрів налаштувань

Згідно з встановленими параметрами, стерео зображення фігури виглядає дещо інакше, змінюючи візуальні ефекти та глибину сприйняття сцени, що надає більш реалістичний та занурювальний досвід користувачеві.

A screenshot of a computer

Description automatically generated

Рисунок 4.2 – Вигляд Ding-Dong Surface зі стандартними параметрами

A screenshot of a computer

Description automatically generated

Рисунок 4.3 – Вигляд Ding-Dong Surface зі зміненними параметрами

HTML-елемент <audio> дозволяє не тільки керувати відтворенням аудіо-доріжки, але й забезпечує додаткові можливості: перемотування, регулювання гучності та зупинка/продовження відтворення. Це робить його зручним інструментом для інтерактивного керування звуковим контентом.

A long line on a white surface

Description automatically generated

A long line on a white background

Description automatically generated

Рисунок 4.4 – HTML-елемент audio зі звуковою доріжкою та без

Також було реалізовано чекбокс для керування станом звукового фільтра: його можна вмикати та вимикати. Залежно від стану фільтра можна помітити зміни у звучанні аудіо, що дозволяє користувачеві гнучко налаштовувати звукове оточення для досягнення бажаного ефекту.

A black text on a white background

Description automatically generated

Рисунок 4.5 – Чекбокс із функціоналом увімкнення/вимкнення фільтру

1. **Код програми**

let context, audio, src, highshelf, spatial;

/\* Initialize the Audio context. Called from init() \*/

function initAudioCtx() {

audio = document.getElementById('audio');

audio.addEventListener('play', () => {

if (!context) {

context = new AudioContext();

src = context.createMediaElementSource(audio);

spatial = context.createPanner();

highshelf = context.createBiquadFilter();

src.connect(spatial);

spatial.connect(highshelf);

highshelf.connect(context.destination);

highshelf.type = 'highshelf';

highshelf.frequency.value = 8000;

highshelf.gain.value = 15;

context.resume();

}

})

audio.addEventListener('pause', () => {

console.log('pause');

context.resume();

})

const highshelfEnabled = document.getElementById('highshelf');

highshelfEnabled.addEventListener('change', function () {

if (highshelfEnabled.checked) {

spatial.disconnect();

spatial.connect(highshelf);

highshelf.connect(context.destination);

} else {

spatial.disconnect();

spatial.connect(context.destination);

}

});

audio.play();

}