

**МІНІСТЕРСТВО КУЛЬТУРИ УКРАЇНИ
НАЦІОНАЛЬНА АКАДЕМІЯ КЕРІВНИХ КАДРІВ
КУЛЬТУРИ І МИСТЕЦТВ
НАВЧАЛЬНО-НАУКОВИЙ ІНСТИТУТ ПЕРФОМАТИВНИХ
МИСТЕЦТВ
Кафедра музичного мистецтва**

КВАЛІФІКАЦІЙНА РОБОТА

на здобуття освітнього ступеня «Магістр»

**на тему: «АРАНЖУВАННЯ МУЗИКИ В ІНТЕРАКТИВНИХ
ІГРОВИХ СЕРЕДОВИЩАХ: КОМПОЗИЦІЙНО-
ТЕХНОЛОГІЧНІ АСПЕКТИ».**

Виконав: здобувач II курсу магістратури,
групи МММ 23-24, спеціальності
025 «Музичне мистецтво»
Томашевський Кирило Святославович

Керівник: канд. мистецтвознав., доц. каф.
музичного мистецтва НАКККіМ, Дьяченко
Володимир Валерійович

Рецензент: канд. мистецтвозн., доц. каф.
муз. мист. естради Акад. мистецтв
ім. П. Чубинського, Левко Вероніка
Іванівна

Допустити до захисту
Протокол засідання кафедри
музичного мистецтва №__
від «__» _____ 2025 р.

Завідувач кафедри _____
(_____)

КИЇВ – 2025

АНОТАЦІЯ

У кваліфікаційній роботі досліджено композиційно-технологічні засади створення адаптивного музичного оформлення для інтерактивних ігрових середовищ. На основі аналізу історичних передумов, еволюції музики у відеоіграх та сучасних теоретичних моделей (vertical layering, horizontal re-sequencing, параметричні та стано-керовані системи) обґрунтовано необхідність інтегрованого підходу, що поєднує методи композиції, звукорежисури та технічної імплементації аудіо.

У роботі розглянуто технологічні можливості провідних ігрових рушіїв (Unity, Unreal Engine) та middleware-платформ (Wwise, FMOD), визначено їх роль у реалізації адаптивної музики. Окрему увагу приділено етапам продакшну: створенню музичних модулів, роботі з цифровими аудіостанціями (DAW), семплерними бібліотеками, запису живих інструментів, мікшуванню й мастерингу інтерактивного аудіо.

Практична частина містить концепцію, створення та технічну реалізацію адаптивного музичного оформлення для ігрового демо-проєкту, що демонструє застосування розглянутих методів у реальному гейм дизайнерському контексті. Проведений аналіз підтверджує ефективність композиційно-технологічного підходу для формування органічного, змінного та емоційно достовірного музичного супроводу в інтерактивних медіа.

Ключові слова: *адаптивна музика, інтерактивне аудіо, відеоігри, vertical layering, horizontal re-sequencing, Wwise, FMOD, Unity, Unreal Engine, музичний продакшн.*

ABSTRACT

The thesis explores the compositional and technological foundations of adaptive music design for interactive game environments. Based on an analysis of the historical development of video game music and modern theoretical models - such as vertical layering, horizontal re-sequencing, parametric systems, and state-driven music structures - the research substantiates the need for an integrated approach that combines composition, sound design, and technical implementation.

The study examines the technological capabilities of leading game engines (Unity, Unreal Engine) and audio middleware tools (Wwise, FMOD), highlighting their importance for creating adaptive and dynamically responsive music systems. Special attention is given to production workflows: developing musical modules, working with digital audio workstations (DAWs), sample libraries, recording live instruments, mixing, and mastering interactive audio.

The practical section presents the concept, creation, and technical implementation of an adaptive musical score for an game demo project, demonstrating how theoretical principles can be applied in a real game development context. The conducted analysis confirms the effectiveness of the compositional-technological approach in producing cohesive, variable, and emotionally immersive music for interactive media.

Keywords: *adaptive music, interactive audio, video games, vertical layering, horizontal re-sequencing, Wwise, FMOD, Unity, Unreal Engine, music production.*

ЗМІСТ

ВСТУП.....	5
РОЗДІЛ 1. ТЕОРЕТИКО-ІСТОРИЧНІ ЗАСАДИ АДАПТИВНОГО МУЗИЧНОГО ОФОРМЛЕННЯ ІГРОВИХ СЕРЕДОВИЩ.....	9
1.1. Ретроспектива музичного оформлення в інтерактивних ігрових середовищах.....	9
1.2. Теоретичні засади адаптивної музики в інтерактивних медіа.....	20
1.3. Продакшн адаптивної музики в інтерактивних медіа.....	34
РОЗДІЛ 2. ТЕХНОЛОГІЧНІ АСПЕКТИ ІМПЛЕМЕНТАЦІЇ АДАПТИВНОЇ МУЗИКИ В ІГРОВИХ РУШІЯХ.....	44
2.1. Огляд сучасних ігрових рушіїв та middleware (на прикладі Unity/Unreal Engine, Wwise/FMOD).....	44
2.2. Методи та інструменти технічної імплементації адаптивного музичного контенту.....	54
2.3. Композиційні стратегії створення адаптивних музичних структур для ігор.	69
РОЗДІЛ 3. ПРАКТИЧНА РЕАЛІЗАЦІЯ ТА АНАЛІЗ АДАПТИВНОГО МУЗИЧНОГО ОФОРМЛЕННЯ ДЕМО-ПРОЄКТУ.....	77
3.1. Концептуальне обґрунтування музичного оформлення обраного ігрового демо-проєкту.....	77
3.2. Процес створення та продакшну адаптивного музичного контенту: композиційні та технічні рішення.	81
3.3. Технічна імплементація музики в ігровий рушій та налаштування параметрів адаптивності (FMOD, Unity).....	87
ВИСНОВКИ	92
СПИСОК ВИКОРИСТАНИХ ДЖЕРЕЛ.....	95
ДОДАТКИ	100

ВСТУП

Стрімка еволюція сучасного медіа-простору під впливом інтерактивних технологій зумовлює трансформацію ролі музики як невід'ємної складової аудіовізуального контенту. У контексті розвитку відеоігор, інтерактивних фільмів, VR/AR-проектів та інших мультимедійних форматів, музичне оформлення перестає виконувати виключно фонову функцію. Натомість воно набуває динамічних характеристик: здатності реагувати на дії користувача та відігравати ключову роль у створенні емоційного занурення і формуванні інтерактивного досвіду. Особливої значущості ці тенденції набувають у сфері відеоігор, де складність ігрового процесу вимагає відповідної гнучкості й адаптивності музичного супроводу. Зазначені зміни актуалізують необхідність переосмислення традиційних підходів до аранжування та музичного продакшну. Сучасні композитори й саунд-дизайнери постають перед завданням не лише написання музичного твору з урахуванням його форми та естетичних якостей, але й розробки логіки його поведінки в інтерактивному середовищі. Це передбачає використання спеціалізованого програмного забезпечення, ігрових рушіїв та аудіо-мідлвеарів (спеціальне програмне забезпечення, яке виступає проміжною ланкою між грою (ігровим рушієм) та аудіоконтентом) задля забезпечення негайного відгуку музики на дії гравця, безшовності переходів і підтримання стилістичної цілісності саундтреку за умов варіативності.

Актуальність обраної теми дослідження зумовлена експоненційним зростанням індустрії відеоігор та інтерактивного контенту, де якісне інтегроване музичне оформлення є вагомим чинником комерційного успіху та художньої цінності продукту. Незважаючи на високий попит на фахівців у цій галузі, на академічному рівні все ще бракує системних досліджень, особливо в українському науковому дискурсі, що підкреслює наукову новизну роботи. К. Коллінз у своїй монографії «Game Sound» зазначала, що з розвитком технологій аудиторія дедалі більше втомлюється від типових зациклених саундтреків, тому впровадження динамічної, адаптивної музики стало

необхідністю [13]. Історична ретроспектива також свідчить, що «типова модель» лупів була спричинена технічними обмеженнями ранніх ігор, але вже тоді робилися спроби урізноманітнити звучання. Сучасний розвиток інтерактивної музики є еволюційним продовженням багаторічної практики, збагаченим теоретичним осмисленням. Таким чином, актуальність і наукова новизна теми зумовлені необхідністю інтегрованого підходу, що поєднує традиційні композиційні техніки з сучасними інтерактивними аудіотехнологіями.

Мета дослідження – комплексне вивчення композиційно-технологічних методів адаптивного аранжування та технічної імплементації музичного матеріалу для створення ефективного й художньо виразного музичного оформлення в інтерактивних ігрових середовищах. Під композиційно-технологічними методами розуміються способи побудови музичного матеріалу для інтерактивного середовища: поділ музики на шари (vertical layering), структуризація у фрагменти для їх плавного перемикання (horizontal re-sequencing), модульність, варіативність, використання лейтмотивів та інших прийомів, що дозволяють музиці змінюватися залежно від дій гравця. Під технічними методами маються на увазі практичні засоби створення та впровадження музики: робота в цифрових аудіостанціях (DAW), застосування семплерних бібліотек і синтезаторів, мікшування й обробка звуку, а також інтеграція музичного матеріалу в ігрові рушії та аудіо-middleware (Wwise, FMOD) для реалізації адаптивності. Сукупність цих методів забезпечує створення ефективного та художньо виразного музичного супроводу в інтерактивних ігрових середовищах.

Об'єкт дослідження – музика в інтерактивних ігрових середовищах.

Предмет дослідження – композиційні та технологічні методи адаптивного аранжування і технічної імплементації музичного матеріалу в ігрових рушіях, що забезпечують динамічну взаємодію музики з ігровим процесом.

Для досягнення визначеної мети у межах кваліфікаційної роботи було визначено такі завдання дослідження:

1. Здійснити історико-теоретичний аналіз становлення і розвитку музичного оформлення в інтерактивних ігрових середовищах, виокремивши ключові етапи еволюції та теоретичні засади адаптивного аудіо.

2. Виявити та систематизувати сучасні композиційні підходи до створення адаптивних музичних структур, що застосовуються у розробці музики для інтерактивних ігор (принципи нелінійності, варіативності та реактивності).

3. Проаналізувати технологічні рішення та інструменти (зокрема можливості ігрових рушіїв і аудіо-мідлвеарів), які використовуються для технічної імплементації адаптивного музичного контенту в ігровому середовищі.

4. Розробити концепцію адаптивного музичного оформлення для обраного ігрового демо-проєкту, обґрунтувавши художні та технічні рішення відповідно до специфіки проєкту.

5. Реалізувати практичну імплементацію розробленої концепції, включаючи створення адаптивного музичного контенту та його інтеграцію в ігровий рушій з використанням відповідних технологій.

6. Провести якісний аналіз результатів практичної реалізації, оцінивши ефективність використаних композиційних і технологічних методів з точки зору адаптивності, художньої виразності та технічної доцільності музичного оформлення демо-проєкту.

У роботі застосовано комплекс методів дослідження, зокрема теоретичні: аналіз та систематизація наукової літератури з саунд-дизайну, теорії музики, інтерактивного аудіо; порівняльно-типологічний і порівняльно-технологічний аналіз музичних підходів. Серед емпіричних методів використано практичне моделювання (експериментальне створення адаптивних музичних фрагментів), технічну імплементацію в ігровому середовищі та експертну оцінку результатів.

Теоретичною базою дослідження стали:

1) напрацювання зі звукорежисури (Вініфред Філіпс; «Кембриджський путівник з відеоігрової музики» тощо);

2) інтерв'ю та статті професійної спрямованості (Джо Боссо, EnsiGame; Ouranio Recordings; Джеймс Паркінсон тощо);

3) роботи з психології творчості (Global Step; The Game Audio Co тощо).

4) технічна література (сторінки та інструкції аудіо-продуктів Slate Digital «Virtual Tape Machines», Soundtoys «SpaceBlender», Plugin Alliance «Elysia Alpha Compressor», MasteringBox, AudioEase «Altiverb», Audiokinetic Wwise, FMOD Studio та інших);

5) джерела з композиції, мелодики, гармонії, формотворення та написання музики (Вініфред Філіпс, Карен Коллінз, Тім Саммерс, Octaton, Cambridge Core тощо).

Наукова новизна роботи полягає в обґрунтуванні інтегрованого підходу до створення музики для інтерактивного середовища, який поєднує традиційні техніки композиції та аранжування із сучасними технологіями динамічного аудіо.

Практичне значення одержаних результатів визначається можливістю використання розробленої методології як практичних рекомендацій для композиторів і саунд-дизайнерів, що працюють у сфері ігрової індустрії, кібермистецтва та інтерактивного медіаконтенту, а також в освітньому процесі композиторів, музичних продюсерів, звукорежисерів.

Апробація результатів та публікації. Томашевський К. С. Продакшн ритм-секції в нео соулі: технології та творчі підходи на прикладі альбому «origin» виконавця Jordan Rakei» / Культура і мистецтво: сучасний науковий вимір : матеріали ІХ міжнар. наук. конф. Молодих вчених, аспірантів та магістрів, 6 лист. 2025 р. М-во культ. України; Нац. акад. кер. кадрів культ. і мистец. Київ: НАКККиМ, 2025.

Структура кваліфікаційної роботи зумовлена метою і завданнями дослідження. Робота складається зі вступу, основної частини (три розділи, що містять вісім підрозділів), висновків, списку використаних джерел (56 позицій). Загальний обсяг роботи - 108 сторінок, з них основний текст займає 94 сторінки.

РОЗДІЛ 1. ТЕОРЕТИКО-ІСТОРИЧНІ ЗАСАДИ АДАПТИВНОГО МУЗИЧНОГО ОФОРМЛЕННЯ ІГРОВИХ СЕРЕДОВИЩ

1.1. Ретроспектива музичного оформлення в інтерактивних ігрових середовищах.

Перші покоління відеоігрової музики формувалися в умовах суворих апаратних обмежень, що прямо визначали характер звучання та композиційні підходи. Одним із ключових факторів була обмежена поліфонія звукових чипів, тобто кількість доступних звукових каналів. Аркадні автомати кінця 1970-х - початку 1980-х зазвичай підтримували від одного до трьох одночасних голосів. Наприклад, домашня консоль Atari 2600 (1977) мала звуковий чип ТІА лише з двома незалежними каналами, тож композиторам доводилося розташовувати мелодію та ритмічний супровід дуже економно у фактурі. Подібно, ранні аркадні ігри часом узагалі відтворювали лише один звуковий потік. У більш розвинутій Nintendo Entertainment System (NES, 1983) апаратні можливості були ширшими: її звуковий процесор (Audio Processing Unit, APU) забезпечував п'ять каналів – два канали для прямокутних хвиль (мелодії/гармонії), один для трикутної хвилі (найчастіше бас), один для білого шуму (перкусійні ефекти) та один канал дельта-модуляції для відтворення низькоякісних семплів. Це дало змогу дещо розширити палітру звуку, проте все одно не дозволяло повноцінної багатоголосності, характерної для популярної музики того часу. Так, NES фактично міг відтворювати лише до трьох тонів одночасно (не рахуючи шумових ефектів), оскільки один із п'яти каналів був шумовим, а інший - монофонічним семплером [13].

Другим істотним обмеженням був мінімальний обсяг пам'яті для аудіо-інформації. Ранні ігри постачалися на картриджах обсягом лише 16–32 КБ, де потрібно було розмістити все: код гри, графіку, звукові ефекти і музику. Для порівняння, у персональних комп'ютерах тієї епохи оперативна пам'ять часто вимірювалася одиницями чи десятками кілобайтів, і навіть зберігання коротких

мелодій займало цінні байти. Внаслідок цього музичні фрагменти були гранично короткими, зазвичай це були мелодійні фрази від кількох секунд до півхвилини, які циклічно повторювалися протягом усього рівня чи гри. Такий циклічний принцип став фактично стандартом: нескінченне повторення короткої теми було вимушеною необхідністю, продиктованою пам'яттю, але згодом перетворилося на особливу естетичну рису ігрової музики [13].

Третім базовим обмеженням була примітивна темброва палітра ранніх звукових схем. Більшість чипів генерували елементарні хвильові форми: прямокутні (імпульсні) коливання, трикутні й пилкоподібні хвилі; інколи додавався канал шуму. Наприклад, згаданий звуковий модуль NES надавав два імпульсні осцилятори з регульованою широтою імпульсу, один генератор трикутної хвилі та один шумовий генератор [13]. В аналогічних системах (Commodore 64, Sega Master System тощо) набори хвиль дещо різнилися, але загалом тембри залишалися «електронними» і синтетичними - без можливості реалістичного відтворення акустичних інструментів. Це породило характерний стиль chiptune, для якого властива темброва простота, компенсована виразною мелодикою та енергійним ритмом.

Таким чином, кількість звукових каналів, обсяг пам'яті та набір доступних тембрів стали базовими параметрами, що визначили специфічну логіку ігрової музики 1970–1980-х років. Апаратура диктувала умови: композитори мусили працювати в умовах жорстких лімітів, що сприяло народженню нових творчих прийомів. Замість повноцінних аранжувань - моно- чи двоголосні мелодії; замість тривалих розгорнутих форм - короткі циклічні зациклення; замість поліфонії - імітація акордів швидкими арпеджіо (послідовне дуже швидке програвання тонів акорду на одному каналі) [13]. Усе це було прямим наслідком апаратних обмежень, що стали передумовою для перших ігрових саундтреків.

У відповідь на перелічені технічні обмеження композитори відеоігор виробили унікальні стратегічні підходи до написання музики, багато з яких стали прообразами сучасних інтерактивних методів. Передусім, музика отримала модульну структуру: вона складалася з автономних фрагментів (модулів),

здатних повторюватися і програватися залежно від ігрової ситуації. Такий підхід був логічним виходом, оскільки неможливо записати тривалий саундтрек, натомість створювалися стислі сегменти, що витримували багаторазове повторення без надмірної набридливості.

Однією з головних вимог було, щоб коротка музична петля не набридала гравцю, навіть лунаючи десятки разів. Для цього використовувалися мелодично виразні, але стримані за розвитком теми, іноді з елементами варіативності. Композитор і теоретик Тім Саммерс зазначає, що повторювані ігрові теми цього періоду можна розглядати як своєрідний «модульний мінімалізм», оскільки вони складаються з простих мотивів, що переставляються й нашаровуються подібно до мінімалістичних композицій [43]. Звісно, таке трактування підкреслює естетичний бік явища: примусова циклічність перетворилася на художній принцип, що гармонує з інтерактивністю гри.

Поєднання музики і звукових ефектів було ще одним викликом. Через малу кількість каналів музика і звукові ефекти конкурували за ресурси. Нерідко доводилося жертвувати одним заради іншого: у багатьох іграх, коли лунав гучний ефект (постріл, вибух), один з музичних каналів вимикався. Це спостерігалось навіть у знаменитому саундтреку Кодзі Кондо до «Super Mario Bros.» (1985): при стрибку Маріо звук стрибка забирає канал у мелодії, тимчасово її перериваючи [37]. Композитори винайшли різні прийоми, щоб зробити такі вимушені паузи мінімально шкідливими для музики. Зокрема, Кондо оптимізував аранжування так, що головна мелодія дублюється з басовою лінією у гармонічному плані, а звук стрибка підібраний по висоті до тональності твору. Отже, коли він «витісняє» мелодію, загальна гармонія не руйнується [там само]. По суті, звуковий ефект інтегрується в музичну тканину як ще один голос, зберігаючи цілісність звукової картини гри.

Інші стратегії включали використання окремого каналу виключно під звуки або музику. Деякі ігри йшли шляхом надання гравцю вибору між музикою та ефектами - наприклад, в окремих версіях «Chronicles of Omega» або «Delta» для Commodore 64 пропонувалося вимкнути музику, щоб чути лише ефекти [13].

Таке рішення підкреслює, наскільки цінним був кожен канал звуку. Ще одним трюком було «екстремальне мультиплексування» нот - техніка, коли один голос по чергово, дуже швидко, програє різні партії, створюючи ілюзію кількох одночасних ліній [там само]. Так діяв, скажімо, відомий композитор Роб Габбард на Commodore 64, аби скласти враження, ніби чип SID видає більше трьох голосів одночасно. Подібно, на NES і C64 практикували арпеджіо-акорди: замість справжнього акорду відбувалося блискавичне програвання його складових звуків одним каналом, що сприймається вухом як дзвінкий «акордовий» ефект. Ці прийоми вимагали від композиторів неабиякої кмітливості і програмістської співпраці, адже музика часто кодувалася вручну в вигляді драйверів - спеціальних програм для звукового чипа [там само]. У результаті технічні можливості трансформувалися в музичні засоби: обмеження стимулювали творчість, породжуючи нові методи композиції, що зберігали актуальність і надалі.

Врешті, важливою характеристикою стала циклічна структура саундтреку, прив'язана до ігрових станів. Оскільки ігри побудовані на повторенні схожих дій (рівнів, раундів), музика теж набула повторювано-модульного характеру. Кожен рівень чи ситуація отримували свій короткий музичний мотив, який грав по колу, поки стан не зміниться. Така адаптивна прив'язка музики до геймплею була прообразом подальших динамічних саундтреків. По суті, вже у 1980-х склалася система простих «музичних станів». Наприклад, в одній ситуації звучить спокійна тема, при появі ворогів - тривожний мотив, при бою з босом - прискорена ритмічна тема тощо. Хоч перемикання між ними було ще дискретним (неплавним), сам підхід заклав основу для сучасних технологій *horizontal resequencing* (горизонтальної секвенції) - переключення між підготованими фрагментами музики, - та *vertical layering* (вертикального нашаровування) - додавання/прибирання шарів в багат шаровій композиції залежно від ситуації [13]. Про ці концепції докладніше йтиметься нижче; важливо підкреслити, що їхні витoki - у вимушеній модульності ранніх ігрових мелодій.

Хоча перші відеоігри (1970-ті роки) були здебільшого німі або обмежувалися короткими звуковими сигналами, вже наприкінці 1970-х з'являються новаторські приклади постійного музичного супроводу, інтегрованого з геймплеєм. Одним із піонерів стала аркадна гра «Space Invaders» (Taito, 1978). Її саундтрек складався всього з чотирьох нот, що безперервно повторювалися остинатним басовим мотивом. Цей простий мотив поступово прискорювався в міру того, як інопланетяни наближалися до гравця. Вперше у відеоігрі музика динамічно реагувала на події гри: пришвидшуючись, вона посилювала напругу і тиск на гравця. Продемонструвавши, що ігровий звук може бути чимось більшим, ніж простим заповненням тиші, «Space Invaders» наблизила музику відеоігор до сфери мистецтва» [13]. Таким чином, «Space Invaders» заклала фундамент ідеї інтерактивної музики: музика не лише фоново грала, а й слугувала геймплейним індикатором (прискорення темпу попереджає про зростання небезпеки).

У 1980 році інша знакова аркада «Pac-Man» (Namco), яка продемонструвала підхід «музичних сигналів». Основний геймплей Pac-Man взагалі не має фонового треку (лише звуки «вживання» точок), проте короткі мотиви-фанфари використовуються для позначення ключових подій чи переходів. Наприклад, початок рівня супроводжується веселою фанфарою, що задає настрій; у момент коли «Пак-Мен» з'їдає спеціальний «енерджайзер» і сам починає переслідувати привидів, миттєво вмикається інший короткий музичний фрагмент (тривожний сигнал), який триває поки діє цей режим. Як тільки ефект минає, музика повертається до попереднього стану (або тиші). Ці звукові маркери виконують роль індикаторів стану гри, оперативно змінюючи атмосферу гри відповідно до дій гравця. Попри граничну простоту, такий підхід заклав ідею контекстуальної музичної реактивності: навіть короткий звук може нести інформацію про зміну ігрового стану.

Наступного року з'являється «Donkey Kong» (Nintendo, 1981) - гра, важлива не лише як дебют Маріо, але і як приклад розвитку музичних рішень. У «Donkey Kong» кожен етап рівня починається з кількасекундної мелодії-

заставки, яка потім переходить у циклічний бекграунд (ритмічний *ostinato*) під час ігрового процесу. При переході на новий рівень або при виграші звучить інша коротенька мелодія, що «розділяє» ігрові етапи. Через обмеження пам'яті ці теми надзвичайно лаконічні, але їх вистачає, щоб надати грі відчуття драматургії: гравець чує початок пригоди, кульмінації та завершення рівня у вигляді впізнаваних звукових заставок. Таким чином, *Donkey Kong* продемонструвала, як декілька секунд музики можуть виконувати структурну функцію, сигналізуючи про прогрес у грі.

Варто згадати і про «*Super Mario Bros.*» (NES, 1985), гру, що піднесла інтерактивну музику на новий рівень. Композитор Кодзі Кондо створив шість основних музичних тем, кожна з яких відповідала певному середовищу або ситуації: основна тема для надземних рівнів, окрема тема для підземель, підводний вальс, «марш замку», мелодія зірки (непереможності) та фінальна фанфара [37]. Музика була нерозривно пов'язана з ігровим процесом і його ритмом. Кондо свідомо синхронізував музичний характер з «відчуттям гри»: коли гравець отримує зірку та стає невразливим, грає швидка радісна мелодія; коли час на рівні добігає кінця, основна тема раптово подвоює темп, сигналізуючи, що треба поквапитися [там само]. Ці прийоми (зміна темпу і тональності в реальному часі) зробили музику активним елементом геймплею, що керує емоціями гравця. Фактично, «*Super Mario Bros.*» став зразком того, як динамічна адаптація музики збагачує ігровий досвід, і вплинув на безліч майбутніх проєктів.

Підсумовуючи, уже в ранніх 8-бітних іграх можна побачити зародки майже всіх основних форм інтерактивної музики:

- 1) Безперервна циклічна музика, що реагує на інтенсивність геймплею («*Space Invaders*», прискорення темпу під натиском ворогів) [34].
- 2) Контекстуальні музичні вставки для різних станів гри - («*Rac-Man*», фанфари для початку рівня і режиму переслідування).
- 3) Структурні музичні розділювачі і теми рівнів - (*Donkey Kong*, окремі мелодії на початок та кінець етапу).

4) Адаптивна зміна параметрів музики в реальному часі - («Super Mario Bros.», прискорення музики при низькому таймері, накладання звуку стрибка як частини музики) [8; **Ошибка! Источник ссылки не найден.**].

5) «Бойові» чи «boss» теми - багато ігор кінця 80-х запровадили спеціальні треки для сутичок з босами або інших напружених моментів, які автоматично увімкалися при вході в зону босса та вимикалися після перемоги або відступу гравця [37]. Це прямий предок сучасної технології горизонтальної секвенції, де фрагменти саундтреку переключаються залежно від ігрового стану.

Усі ці підходи виникли як відповідь на технічні обмеження та потребу підсилити ігровий процес. Вони ж стали основою для подальших розробок у сфері інтерактивної музики.

Цікаво, що еволюція музики відеоігор у 1970–80-х роках відбувалася паралельно до ширших культурних тенденцій у музиці, що забезпечило теоретичні витоки та естетичне обґрунтування для ігрової аудіо-стилістики.

По-перше, у академічній музичній сфері саме в ті десятиліття утвердився напрям мінімалізму (композитори Стів Райх, Філіп Гласс, Террі Райлі та ін.). Мінімалізм відзначається використанням повторюваних структур, остинатних ритмів і поступових змін упродовж тривалого часу. Як відзначав Стів Райх, його цікавили «процеси, які можна почути у звучанні музики», а повторення виступало формою змінності [46]. В мінімалістичних творах, на відміну від традиційних симфоній, відсутня лінійна драматургія з чітким початком-кульмінацією-кінцем; натомість слухач занурюється у «гіпнотичний» циклічний процес. Ця естетика напрочуд співзвучна з природою ранньої ігрової музики: короткі циклічні петлі, які можуть лунати нескінченно, створюючи відчуття «нескінченного теперішнього» в ігровому світі. Якщо у традиційній музиці повтор багаторазово одного і того ж вважався спрощенням, то і мінімалісти, і композитори ігор перетворили повторення на чесноту - спосіб занурення слухача. Деякі дослідники прямо порівнюють «петлі» 8-бітної музики з мінімалістичними техніками. Так, Тім Саммерс підкреслює, що циклічність і лаконічність ігрових саундтреків 80-х - це не просто технічна необхідність, а й

художній метод, який можна назвати «модульним мінімалізмом», адже він дає слухачеві (гравцеві) можливість повністю зануритися в повторюваний музичний мотив, подібно до того, як це відбувається при слуханні мінімалізму [20].

По-друге, 1980-ті - це злет електронної популярної музики: синт-поп, електро, техно, хаус, індастріал та інші жанри, що базуються на ритмічній повторюваності і використанні драм-машин та секвенсорів. Танцювальна музика використовує петлі й біт, які можуть гратися довго без перерв, створюючи атмосферу безперервної вечірки. У нічних клубах звучання циклічного біту занурює відвідувачів у транс, стираючи відчуття часу - ніч здається нескінченною, поки грає музика [46]. Аналогічно, у відеогрі фонова мелодія грає доти, доки триває рівень, підтримуючи сталість і ритм процесу. Не випадково між іграми та електронною музикою налагодився двосторонній обмін: вже в 1980-ті діджеї й продюсери почали семплувати звуки і мелодії з аркад (відомий трек «Computer Game» японського гурту «Yellow Magic Orchestra» 1978 року включав звуки аркадної гри і вплинув на ранніх техно-музикантів у США, а композитори ігор надихалися електронними жанрами. Особливо це проявилось в 1990-х, коли такі саундтреки, як «Streets of Rage» (композитор Юзо Косіро), практично відтворювали естетику клубної музики: по суті, гра дарувала підліткам «перший досвід рейву» ще до потрапляння у справжній клуб [56]. Цей взаємовплив легітимізував використання машинних ритмів і семплів у іграх, а також підкреслив, що ігрова музика є частиною ширшої музичної культури, а не ізольованим феноменом.

По-третє, концепція ambient-музики та «фоновості» мистецтва теж стала важливим культурним контекстом. 1978 року Браян Іно випустив альбом «Music for Airports», ввівши термін «ambient» для позначення музики, що «не нав'язується слухачеві, а створює оточуючий звуковий пейзаж». Іно прагнув зробити музику схожою на ландшафт, частиною простору, у якому можна перебувати [26]. Така музика могла грати вічно, повторюючись і змінюючись ледь помітно, і слухач міг або ігнорувати її, або звертати увагу на деталі - обидва підходи легітимні. Ця ідея безпосередньо перегукується з роллю ігрової музики:

саундтрек у грі часто виконує функцію безперервного звукового фону, що супроводжує гравця. Як зазначив сам Браян Іно, його ambient-ландшафти покликані створювати «пейзаж, до якого ти належиш», подібно до «безкінечно повторюваних світів відеоігор», де музика циклічно грає годинами [там само]. Більш того, Іно підкреслював, що «повторення – це різновид змін», маючи на увазі, що у тривалому повторі поступово змінюється сприйняття та дрібні нюанси. У іграх ми бачимо той самий принцип: хоч мелодія повторюється, ігрові події накладають на неї нові відтінки (звуки пострілів, зміни темпу, переходи на інший трек при зміні рівня тощо), тож досвід не є статичним.

У підсумку, нескінченна циклічність ранньої ігрової музики була співзвучна часові. З одного боку, вона визначалася вимогами технології, але з іншого - відповідала естетичним пошукам епохи. Академічний мінімалізм утвердив повтор як естетично значущу форму; електронна танцювальна музика продемонструвала, що монотонність біту може захоплювати і переносити в транс; ambient-естетика довела, що музика може бути частиною середовища, непомітною, але впливовою. Ці ідеї створили сприятливий ґрунт для сприйняття ігрової музики не як примітивного «пікання», а як нової форми музичного мистецтва, що оперує процесами, а не творами. Повтор у цьому контексті перестав бути недоліком, натомість став «ритуалом» - як зауважує Саммерс, музика у грі виконує роль ритуального елемента, що повторюється у кожній сесії гри, створюючи знайоме середовище і структуру для дій гравця [44].

Розглянувши історичні та культурні передумови, можна простежити, як ранні рішення переросли в теоретичні концепції інтерактивної музики, що нині активно застосовуються. Дві ключові технології в сучасних інтерактивних медіа - це горизонтальна секвенція (horizontal re-sequencing) та вертикальне нашарування (vertical layering) музики [**Ошибка! Источник ссылки не найден.**; 21].

Горизонтальна секвенція передбачає, що саундтрек складається з наперед написаних музичних фрагментів, які можуть динамічно перемикатися або комбінуватися послідовно залежно від перебігу гри. По суті, це розвиток ідеї

«музичних станів» з ранніх ігор. Наприклад, як у «Mayhem in Monsterland» або інших класичних іграх при появі ворога або босу запускала інша музична тема [44], так і сучасна гра може перемикнути спокійну фонову музику на епічну сцену, коли починається бій. Цей метод зараз реалізується гнучкіше: композитори пишуть кілька варіантів треку (для спокійної фази, напруженої, критичної тощо), а ігровий рушій непомітно «склеює» потрібні сегменти в реальному часі. Однак принцип той самий - музика рухається горизонтально між різними заготовленими відрізками.

Вертикальний леєринг (або вертикальна ре-оркестровка) - це підхід, коли одночасно грає кілька шарів музики, і залежно від контексту деякі з них додаються або прибираються [**Ошибка! Источник ссылки не найден.**]. Таким чином, змінюється насиченість і характер музики без переривання її ходу. Цей метод теж має свої зародки у минулому. Вже у «Super Mario Bros.» можна помітити зародок вертикального принципу: мелодія, бас і ритм були розкладені по окремих каналах, і, наприклад, звук таймера або стрибка фактично поведився як додатковий шар (хоч і короткочасний) поверх основного треку [37]. Пізніше, на апаратурі 16-біт, з'явилися реалізації примітивного леєрингу, скажімо, динамічне загасання/повернення деяких голосів залежно від ситуації. Сучасні ігрові звукові рушії (FMOD, Wwise тощо) дозволяють композитору записати, умовно, оркестровий трек, розкладений на шари (ударні, струнні, електронні ефекти), що можуть увімкнутися всі разом у кульмінаційний момент або зменшуватися до одного-двох інструментів у спокійні фази. Тематика залишається та сама, але її аранжування адаптується до напруги гри. Вертикальне нашарування таким чином вирішує проблему «повторюваності»: музика залишається циклічною, але постійно варіюється за рахунок мікшування шарів.

І горизонтальний, і вертикальний метод були теоретично описані у літературі з інтерактивного аудіо. Зокрема, Вінніфред Філліпс у книзі «A Composer's Guide to Game Music» [**Ошибка! Источник ссылки не найден.**] чітко визначила ці дві стратегії як основні інструменти створення динамічного

аудіо. Важливо підкреслити, що хоча сучасні технології дозволяють більш плавні і складні реалізації (наприклад, зшивання фрагментів відбувається по долям, з врахуванням гармонії, а перехід між шарами - з кросс-фейдами та фільтрами), - концептуально ці прийоми вирости з простих ідей 80-х.

Ще один сучасний термін - «динамічна музика» - використовується для позначення будь-якої музики, що змінюється залежно від гри (включаючи інтерактивну та адаптивну за класифікацією Коллінз [14]. Коллінз у своїй монографії «Game Sound» зазначала, що з розвитком технологій гравці дедалі більше втомлюються від «типової моделі зацикленого відтворення», тож впровадження динамічних саундтреків стало необхідністю [13]. Історична ретроспектива показує, що «типова модель» зацикленої музики була породжена технічними умовами перших ігор, але вже тоді робилися спроби цю модель урізноманітнити. Отже, сучасний розвиток інтерактивної музики - не революційний розрив, а еволюційне продовження багаторічної практики, осмислене і збагачене теорією.

Ретроспективний огляд показує, що музика в інтерактивних медіа (передусім у відеоіграх) пройшла шлях від простих мелодійних циклів, зумовлених апаратними обмеженнями, до складних динамічних саундтреків, які реагують на дії користувача в режимі реального часу. Теоретичні витoki цього явища закорінені як у самій практиці ранніх розробників ігор, так і у ширших музичних тенденціях другої половини ХХ століття. В умовах 1-3-голосої поліфонії та кількасекундних семплів, перші ігрові композитори створили саундтреки, які не тільки функціонували в заданих межах, але й сформували нову естетику повторення, модульності та процесуальності.

Література з цього питання сходиться на тому, що ігрова музика від самого початку була невіддільна від інтерактивності. На відміну від кіномузики, що є лінійною і підпорядкованою фіксованому сюжету, музика в іграх мусила бути готовою до непередбачуваного розвитку подій - і це стало її визначальною особливістю. Ранні технічні рішення - зациклення, перемикання треків, накладання звуків - нині переосмислені у вигляді концепцій *horizontal re-*

sequencing та vertical layering, що дозволяють зберегти внутрішню логіку музики і одночасно підлаштувати її під дії гравця [Ошибка! Источник ссылки не найден.; 21; 44].

Отже, ретроспектива розвитку ігрової музики демонструє, що обмеження 1970–80-х років несподівано стали благодатним ґрунтом для інновацій, які визначили обличчя інтерактивного аудіо. Короткі повторювані мелодії епохи 8-біт не були примітивним курйозом - вони виконували чіткі функції: сигналізували про стани, керували емоційним ритмом гри, створювали відчуття безперервності світу. Це підготувало аудиторію до сприйняття більш складних музичних систем у подальшому. Успадкувавши і розвинувши ці ідеї, сучасні інтерактивні медіа використовують музику як гнучкий, адаптивний компонент ігрового досвіду. Історичні та теоретичні витоки показують: від перших чотирьох нот «Space Invaders» до сучасних оркестрових саундтреків із процедурною генерацією - це логічний шлях еволюції, де кожен крок базувався на попередньому досвіді. Музика в інтерактивних медіа сьогодні - це синтез технології та мистецтва, що продовжує принципи, закладені ще у «золотий вік» аркадної культури [Ошибка! Источник ссылки не найден.] , і водночас використовує потенціал новітніх ідей, коріння яких також сягає минулого.

Таким чином, ретроспективний аналіз підтверджує: теорія інтерактивної музики народилася з практики, а практика, у свою чергу, надихалася теорією - хай іноді й несвідомо. Ранні композитори відеоігор фактично стали мінімалістами та саунд-художниками за необхідністю, а їхні напрацювання лягли в основу сучасного розуміння музики як частини інтерактивного середовища. Це підкреслює значущість дослідження історії ігрової музики: розуміння її витоків дозволяє глибше осягнути принципи, за якими вона функціонує нині, та перспективи її подальшого розвитку.

1.2. Теоретичні засади адаптивної музики в інтерактивних медіа.

Інтерактивна музика кардинально відрізняється від традиційної лінійної музики, що звучить у кіно чи на концерті. Якщо в кіно музика слідує за фіксованим сценарієм та точно синхронізована з подіями фільму, то в відеогрі розвиток подій непередбачуваний, залежить від дій гравця. Тому й музичний супровід має бути гнучким та адаптивним, здатним змінюватися у реальному часі під впливом ігрових подій [1]. На відміну від лінійного «закритого» твору, який має чітко визначений початок, кульмінацію і фінал, інтерактивний саундтрек більше нагадує лабіринт із множиною можливих шляхів розвитку [8]. Композитор не може повністю контролювати, коли саме прозвучить той чи інший фрагмент, замість цього він конструює музичну систему, здатну реагувати на сценарії гри.

Ранні відеоігри, з огляду на технічні обмеження, переважно використовували циклічні музичні фрагменти (лупи), що безперервно повторювалися під час рівня або сцени. Такий підхід забезпечував фон без пауз, але не враховував динаміку геймплею. З розвитком технологій та зростанням очікувань щодо емоційної взаємодії, виникла потреба в адаптивній музиці, яка змінюється синхронно з подіями гри. Як справедливо зазначає дослідниця Карен Коллінз, невідповідний або розірваний музичний супровід здатен порушити занурення гравця: «розрізнена партитура зазвичай призводить до розрізненого ігрового досвіду» [13]. Отже, цілісність аудіовізуального ряду в інтерактивному медіа вимагає спеціальних композиційних підходів.

У фундаменті теорії інтерактивної композиції лежить ідея модульності. Музичний твір конструюється з окремих фрагментів або шарів, які можуть комбінуватися, вмикатися або вимикатися програмним шляхом. На відміну від звичайного аранжування, де композиція фіксується у сталому вигляді, інтерактивне аранжування передбачає, що музика ніколи не матиме єдиного остаточного варіанту звучання - вона існує у стані постійного потоку [**Ошибка! Источник ссылки не найден.**]. Композитор готує набір музичних елементів, а ігровий рушій маніпулює ними: активує та деактивує інструментальні доріжки, перемикає розділи твору, змінює інтенсивність звучання залежно від ситуації.

Такий підхід створює величезний простір для варіативності, дозволяючи саундтреку підлаштовуватися під дії гравця майже так само, як це робить живий акомпанемент під час імпровізованого театру чи перформансу.

Досягнення виразності в інтерактивній музиці спирається також на психологічні засади сприйняття. Музика в грі впливає на емоції, рівень напруги, збудження або спокою гравця, тому динамічна адаптація музики покликана посилювати потрібні відчуття. Сьогодні ігри пропонують набагато складніші сценарії, і музика має більш тонко реагувати на зміни: від переходу локації до розвитку сюжету. Теоретичні засади інтерактивної композиції охоплюють принципи негайного відгуку музики на дії користувача, безшовності переходів та підтримання естетичної цілісності за умов варіативності.

Таким чином, інтерактивне середовище диктує нові правила для композиторів: гнучкість, модульність і реактивність стають ключовими принципами написання музики для відеоігор. Нижче розглянемо конкретні методи аранжування, що реалізують ці принципи, а також проаналізуємо музикознавчі особливості інтерактивних саундтреків та практичні аспекти їх продакшну.

Сучасні адаптивні саундтреки найчастіше будуються на двох базових підходах: вертикальному леєрингу та горизонтальній секвенції (також відомій як горизонтальне ресеквенсування). Обидва методи дозволяють змінювати музику в реальному часі, але різними способами. Як було описано в минулому підрозділі, вертикальний леєринг передбачає накладання декількох музичних шарів одночасно, тоді як горизонтальна секвенція - перемикання між різними музичними фрагментами послідовно. У практиці композиції для ігор ці методи часто комбінуються, забезпечуючи як варіативність текстури, так і гнучку структуру саундтреку [54].

Вертикальний леєринг (vertical layering) полягає в тому, що музичний твір розділяється на паралельні шари (стеми), які синхронно відтворюються у грі. Кожен шар містить окремий елемент композиції, наприклад, ритмічну секцію, гармонічний супровід, мелодію або атмосферні звуки [1].

При одночасному відтворенні всіх шарів ми чуємо повну оркестровку твору, проте гра може у будь-який момент вимкнути або ввімкнути окремі шари. Таким чином один і той самий музичний матеріал адаптується до різних ігрових ситуацій: наприклад, базовий спокійний шар грає постійно, а додаткові шари з'являються при зростанні напруги, наближенні ворогів чи переході до бойової фази. Важливий композиційний прийом тут - створити такий музичний фундамент, який міг би залишатися привабливим і в усіченому вигляді, і в повному аранжуванні. Часто основа композиції робиться гармонійно нейтральною або повторюваною, щоб дозволити накладати різні за настроєм елементи. Як зазначено в одному з досліджень, вертикальний метод зазвичай починається зі «звукоряду, емоційно невизначеного, до якого додаються різні гармонізації, ритмічні фігури та інструментовки, що можуть поступово вводитися або забиратися під час гри» [1]. Це дозволяє одній і тій самій музичній темі викликати різні емоції залежно від комбінації активних шарів. Наприклад, якщо додати дисонуючі гармонії та уривчасті ритми, саундтрек сприймається напруженіше, а повернення до консонантних акордів і рівномірного пульсу повертає відчуття спокою.

При застосуванні вертикального леєрингу композитор повинен подбати про сумісність шарів. Усі можливі комбінації стемів мають звучати збалансовано та музично, щоб гра могла у будь-який момент змішати їх без дисонансів. Це накладає обмеження на аранжування: гармонічні зміни повинні узгоджуватися між шарами, ритмічні малюнки - не конфліктувати, а тембри - доповнювати один одного. Значну роль відіграє мікшування: наперед визначаються гучності кожного шару, щоб при сумуванні не виникало перевантажень чи провалів. Попри ці виклики, вертикальний підхід дає суттєві переваги: по-перше, музика може плавно нарощувати інтенсивність, просто додаючи нові рівні звучання (наприклад, в бою поступово підмішуються додаткові ударні та агресивні синтезаторні партії). По-друге, чергування шарів допомагає уникнути надокучливості лупів, варіюючи комбінації, саундтрек менше приїдається гравцю [1]. Зрештою, вертикальний леєринг дозволяє робити реактивні зміни «на

льоту»: гра може миттєво відгукнутися на дію гравця (наприклад, герой заходить у нову зону, і тут же додається новий інструментальний шар, що символізує цю локацію) **[Ошибка! Источник ссылки не найден.]**.

Горизонтальна секвенція (horizontal sequencing) - це метод, за якого ігровий саундтрек складається з окремих музичних сегментів, між якими відбуваються перемикання відповідно до розвитку подій. Фактично композитор пише серію пов'язаних між собою музичних фрагментів (тематично чи гармонійно), кожен з яких відповідає певному стану гри або етапу рівня. Під час гри система перескакує з одного фрагмента на інший у потрібний момент. На відміну від вертикального підходу, в кожен момент часу звучить тільки один трек, що зменшує навантаження на пам'ять і спрощує мікшування [54]. Класичний приклад горизонтального ресеквенсування - музичний супровід битви з босом, що складається з трьох послідовних тем: вступної напруженої, основної бойової та фінальної переможної [там само]. Коли бос з'являється, запускається перший сегмент, при початку активної фази бою відразу перемикається другий, а після перемоги гра переходить на тріумфальний фінал. Кожен з цих розділів написаний окремо, але всі вони пов'язані спільними мотивами чи гармоніями, щоб перехід сприймався природно і слугував продовженням музичної розповіді.

Горизонтальна структура вимагає ретельного опрацювання переходів між сегментами. Якщо просто різко обірвати одну мелодію і почати іншу, це може вибити гравця з атмосфери (створити відчуття стику). Тому композитори використовують різні прийоми для безшовного з'єднання фрагментів: тональні підготовки, ритмічні заповнення, універсальні закінчення фраз, або програмні кроссфейди - плавні згасання однієї доріжки з одночасним нарощуванням іншої. Іноді між великими сегментами вставляються короткі перехідні фрагменти (так звані стінгери), які маскують зміну теми. У грі ці переходи прив'язуються до певних подій або умов. Наприклад, коли здоров'я персонажа падає нижче певного рівня, наступний такт музики буде останнім для спокійної теми і далі, на початку нового тактового циклу, рушій ввімкне трек напруженої битви.

Важливо, що сегменти можуть повторюватися (лупитися) доки не настане команда на перехід: скажімо, спокійна мелодія дослідження буде грати по колу, поки гравець мирно блукає локацією, але варто зустріти ворогів - і система переключиться на інший музичний блок [1].

Практичне застосування горизонтального методу - структурування саундтреку за сценарієм гри. Композитор може орієнтуватися на сюжетні глави, рівні, стадії ігрового процесу. Кожному з цих етапів відповідає певний музичний розділ. Так, у платформері можуть бути окремі теми для дослідження світу, для напружених моментів переслідування, для боїв з рядовими ворогами та для босів. Впродовж рівня гра багаторазово перемикатиметься між цими темами, але завдання музики - зберегти зв'язну емоційну канву. Перевагою горизонтальної секвенції є чітка відповідність музичних шматків подіям: легше досягти виразної драматургії (кожен сегмент як окремий епізод історії). До того ж в один момент звучить лише одна стереодоріжка, що економить ресурси пам'яті й CPU, що важливо для мобільних або VR-ігор з обмеженими ресурсами [54].

Недолік цього методу - ризик частих різких переходів, якщо події змінюються дуже швидко. Тому горизонтальний підхід найкраще працює у відносно сценарно визначених ситуаціях: сюжетні сцени, зміна рівнів, початок/кінець бою тощо [54]. Для хаотичних же ситуацій (наприклад, відкрите дослідження світу з раптовими подіями) ефективніше поєднувати його з вертикальним леєрингом.

Як вже згадувалося, у більшості сучасних ігор використовують комбінацію вертикального та горизонтального способів [54]. Гібридний підхід дозволяє отримати плюси обох систем: мультитрековість вертикального леєрингу дає плавну варіацію інтенсивності, а послідовність горизонтального методу - чітке відображення змін у наративі. Наприклад, у грі «Hollow Knight» композитор Крістофер Ларкін поєднав обидва прийоми: для кожної локації є своя базова тема (горизонтальна зміна між зонами), а під час бою в межах тієї ж локації в ту саму тему додатково накладаються нові шари, посилюючи драматизм [там само]. У результаті музика непомітно адаптується як до переходу в інше місце, так і до

зміни ситуації в межах одного треку. Інший приклад - Journey від композитора Остіна Вінторі: тут саундтрек складається з окремих музичних розділів для різних етапів подорожі (горизонтальний аспект), але всередині цих розділів реалізовано вертикальне нарощування оркестрових шарів залежно від розвитку емоційної напруги [там само]. Такий багат шаровий підхід дозволив створити відчуття «плинної» музики, що ніби дихає разом із гравцем і реагує на його дії, за що «Journey» неодноразово відзначали критики.

Отже, методи аранжування інтерактивної музики спираються на ідею розбиття твору на гнучкі компоненти - чи то паралельні шари, чи то послідовні сегменти. Головна мета - забезпечити динамічну змінність саундтреку без втрати художньої цілісності. Композитор виступає тут не лише як митець, але й як системний дизайнер, продумуючи всі можливі варіанти розвитку музики.

Адаптивна природа інтерактивних саундтреків впливає на усі аспекти музичної мови: від гармонії до формотворення. Композитори змушені враховувати, що їхній твір звучатиме невизначено довго, можливо уривками або в змінному оркестровому складі, і при цьому має залишатися естетично цілісним. Розглянемо особливості основних музичних параметрів у контексті інтерактивної музики.

У динамічних саундтреках гармонічна структура часто спрощується або робиться циклічною. Це пов'язано з необхідністю поєднувати різні фрагменти та шари. Наприклад, у вертикальному лєсрингу базовий шар зазвичай має сталий гармонічний остов (постійний тональний центр чи остинатний акорд), щоб інші шари могли накладатися без гармонічних конфліктів. Така гармонічна нейтральність дозволяє додавати напругу через додаткові дисонанси на окремих шарах, не змінюючи загальної тональності. З іншого боку, у горизонтальній структурі перехід між сегментами часто відбувається на спільній гармонічній основі: завершальний акорд одного фрагмента готує тональність наступного. Це може бути здійснено через педальну ноту, домінантову гармонію чи модулюючий місток, щоб новий розділ сприймався як логічне продовження попереднього. В цілому, інтерактивна музика тяжіє до модального мислення та

нескладних гармонічних послідовностей, що добре витримують повторення і варіації. Водночас, коли гра вимагає, гармонія може змінюватися: скажімо, для передачі зміни настрою можливе перемикання в паралельний мажор/мінор або зміна ладового забарвлення (наприклад, перехід до лакрійського ладу для тривоги). Одним із прийомів є використання адаптивної гармонії: ігровий рушій може перемкнути набір акордів на інший, коли змінюється ігровий стан. У термінах Карен Коллінз це називається «Variable harmony» - варіативна гармонія, тобто зміна акордових послідовностей, тональності або ладу як засіб адаптації [13].

Мелодична лінія в інтерактивній композиції повинна бути достатньо гнучкою, щоб її можна було переривати чи повторювати без втрати музичного сенсу. Це ставить композитора перед дилемою: з одного боку, яскрава мелодія робить саундтрек упізнаваним і емоційним, з іншого – якщо вона занадто розгорнута, її важко інтегрувати в циклічну або розгалужену структуру. Тому часто використовують мотивну роботу та техніку лейтмотивів. Невеликі мелодичні фрази, які легко варіювати, стають будівельними блоками адаптивної музики. Наприклад, у саундтреку «Ori and the Blind Forest» Гарета Кокера головна тема героя з'являється у різних треках у зміненому вигляді, що надає цілісності, але її можна вводити в різних контекстах, не чекаючи повного розвитку довгої мелодії. Мелодії в інтерактивній музиці часто мають лупову структуру. Вони написані так, щоб їх кінець природно зливався з початком, дозволяючи повторюватися нескінченно. Інший прийом - мелодичні каденції у потрібних місцях: композитор закінчує фразу на стійкому ступені або паузі, де гра може бути перервана або переключена на інший сегмент без різкого обриву. З огляду на це, мелодика інтерактивних саундтреків інколи менш складна і більш фрагментована, ніж у лінійній симфонічній музичній формі. Однак це компенсується частим використанням мелодичних остинато, вокальних елементів чи ембієнтних мотивів, які створюють атмосферу без домінування над увагою гравця. За необхідності, коли подія гри вимагає акценту, музика може висунути на перший план чітку мелодію (наприклад, фанфари перемоги або

трагічну тему в кульмінаційний момент), але такі епізоди теж проектуються як тригери - спеціальні точки, де музична тема вступає і може завершитися до повернення основного фону.

Ритмічна організація інтерактивної музики, як правило, підпорядкована вимозі синхронізації ігрових подій та плавності переходів. У більшості адаптивних саундтреків композитори дотримуються фіксованого темпу і метру в рамках одного музичного системного модуля, тому що змінювати темп в реальному часі складно технічно і може бути помітно для слухача. Натомість темп може виступати змінним параметром між різними станами гри. Класичний приклад - прискорення музики при появі небезпеки або при обмеженні часу (як у «Super Mario Bros.», де прискорення темпу підганяє гравця, коли спливає час рівня). Таке «variable tempo» рішення також входить до переліку варіативних методів за Коллінз [13]. Проте в межах одного музичного фрагмента темп зазвичай сталий, а зміни ритмічної інтенсивності досягаються іншими засобами: згущенням фактури (більше дрібних нот і ударів на одиницю часу), переходом від розрідженого аранжування до щільного. Ритмічні малюнки в інтерактивній музиці часто циклічні, побудовані на повторюваних пульсуючих фігурах, які можуть нескінченно продовжувати гру. Завдяки цьому, якщо гравець затримається на одному місці, музика здатна «дихати» разом із гравцем, не збиваючись. Цікаво, що іноді композитори свідомо уникають яскраво вираженого ритмічного пульсу в ембієнтних треках, аби ці треки легше змішувалися з іншими і не привертали зайвої уваги. Коли ж потрібна реакція на дію - наприклад, початок битви - музика може відразу увійти на сильну долю такту з чітким ритмом, щоб користувач підсвідомо відчув зміну (так званий музичний сигнал до дії). Щодо метру, частіше використовуються прості розміри (4/4, 3/4), оскільки вони універсальні для синхронізації; складні розміри можуть застосовуватися для створення специфічної атмосфери рівня, але вимагають ретельної координації переходів (можливо, через вставні такти при зміні сегментів).

Традиційні музичні форми (сонатна, варіації, рондо тощо) погано накладаються на нелінійну природу ігрового процесу. Натомість форму інтерактивної музики можна описати поняттями відкритої форми і розгалуженої форми.

Відкрита форма означає, що твір не має наперед визначеної структури розвитку - він скоріше набір модулів, які можуть слідувати в різному порядку або повторюватися скільки треба.

Розгалужена форма (branching form) - це структура саундтреку, що має кілька альтернативних шляхів розвитку залежно від вибору гравця або стану гри. Наприклад, в рольовій грі музика може мати спільний початок, але далі розгалужується на 2-3 варіанти розвитку (через систему горизонтальних гілок), відповідно до того, який поворот сюжету відбувся. Композитор при цьому пише кілька версій продовження композиції, і гра обирає потрібну в реальному часі.

У науковій літературі виділяються навіть мікроформальні варіації, які притаманні інтерактивній музиці: зміни тембру, динаміки, фактури без зміни основного тематичного матеріалу. Коллінз називає їх «micro-variability». Наприклад, варіативна оркестровка (вмикання/вимикання інструментів), варіативне мікшування (зміна балансу), варіативна динаміка (автоматичне коригування гучності) тощо [13]. Усі ці способи дозволяють підтримувати інтерес до музики, не змінюючи її радикально щоразу.

Отже, форма інтерактивного твору - це, по суті, алгоритм можливих варіантів: саундтрек існує як набір правил, за якими розгортається музика (повторюється, переходить, нашаровується), а не як фіксована послідовність тактів. З точки зору музикознавства, це наближує інтерактивну музику до ідеї алеаторики та модульності, що розроблялися ще в академічних експериментах ХХ століття, але тут вони реалізовані у прикладній формі через технології.

Підсумовуючи, адаптивна музика зберігає основні елементи музичної тканини - мелодію, гармонію, ритм - але використовує їх гнучко й варіативно. Гармонія спрощується для сумісності шарів, мелодика дробиться на мотиви для повторюваності, ритміка стабілізується темпово, а форма стає нелінійною та

алгоритмічною. При цьому мета залишається традиційною: посилити емоційний вплив і виразність гри. Різниця в тому, що до звичних виражальних засобів додається новий вимір - інтерактивність, тобто залежність музики від дій користувача. Саме він є визначальним фактором аналізу саундтреків інтерактивних медіа.

Відеоігри як медіа не обмежують композиторів у виборі музичного стилю. Навпаки, тут співіснує і переплітається величезне розмаїття жанрів. Вибір стилістики залежить від жанру гри, сетингу, аудиторії та художнього бачення розробників. Розглянемо основні тенденції застосування різних музичних стилів у сучасних іграх, зокрема оркестрового, електронного, ембієнтного та гібридного.

Жанр симфонічної оркестрової музики міцно утвердився у відеоіграх, особливо в масштабних AAA-проектах (високобюджетних відеоіграх, які створюються великими студіями з великими командами, значними фінансовими ресурсами та тривалими циклами розробки). Сьогодні повноцінний оркестр у саундтреку гри став фактично стандартом для блокбастерів, зрівнявшись за якістю з голлівудськими фільмами. Оркестрова музика надає іграм відчуття епічності, глибини та емоційної ваги. Наприклад, саундтреки до серій «Halo», «The Elder Scrolls», «God of War» виконані симфонічними оркестрами і є невід'ємною частиною їх успіху [47]. Жива оркестровка збагачує ігровий світ нюансами динаміки та тембрів, що особливо важливо для фентезі, пригодницьких та кінематографічних ігор. Оркестровий стиль часто використовується для музичного зображення великих просторів, битв, героїчних моментів - всюди, де потрібен широкий емоційний діапазон.

Важливо, що оркестрова музика у іграх теж підкоряється інтерактивності: записи оркестру розбиваються на окремі теми (струнні, духові, хор тощо) для вертикального леєрингу, або пишуться короткими секціями для горизонтального перемикання. Популярність оркестрового підходу привела навіть до появи концертів і турне, де симфонічні оркестри виконують музику з ігор. Це свідчить про визнання цього жанру широкою аудиторією. У сучасних умовах технічних

можливостей практично немає перешкод для використання живого оркестру: багато студій співпрацюють зі знаменитими оркестровими колективами, записуючи музику в студіях типу Abbey Road чи Air Studios, щоб досягти максимальної якості звучання. Це відображено і у відзнаках: ігрові саундтреки (наприклад, «Baba Yetu» з «Civilization IV» або музика з «Journey») отримували престижні премії, а у 2023 році вперше музика з відеогри здобула Греммі, що підкреслює зростання ролі оркестрової музики в індустрії.

Електронні ж стилі мають давню історію в іграх - від самих початків (чіптюн-звуки 8-бітових консолей) до сучасних електронних жанрів (EDM, техно, саунд-дизайнерські ембієнтні полотна). Електронна музика надзвичайно гнучка за настроєм і фактурою, тому широко використовується для створення атмосферних фонів, науково-фантастичних звукових ландшафтів, хорору чи футуристичних екшенів. Синтезатори дозволяють генерувати унікальні звуки, що виходять за межі можливостей акустичних інструментів. Саме тому у sci-fi іграх або кіберпанкових світах часто домінує електроніка. В сучасних трендах електронна музика в іграх не поступається оркестровій: «DOOM» (2016) з саундтреком Міка Гордона - яскравий приклад жорсткого електронно-індустріального саунду, що ідеально підкреслює інтенсивний геймплей. Цей саундтрек використовує важкі гітарні рифи, агресивні синтезаторні баси і перкусію, зміковуючи елементи industrial, metal і EDM в єдиний потік енергії [34]. Головне тут - створення потужного драйву, який би синхронізувався з діями гравця (в «DOOM» інтенсивність музики зростає при появі ворогів і стихає після очищення арени, тому гравець буквально відчуває ритм бою).

Але електронна музика не обмежується екшеном: у інді-іграх популярні ambient-саундтреки, де за допомогою синтетичних падів, текстур і мінімалістичних мотивів вибудовується медитативна атмосфера. Електроніка чудово підходить для пазлів, симуляторів, стратегії, де потрібен ненав'язливий фон, що не відволікає від роздумів. У цілому, електронна музика забезпечує різноманіття настроїв: від спокійної ембієнтної медитації до адреналінового біта

- тому її елементи присутні майже в усіх жанрах і нерідко поєднуються з оркестровими (створюючи стилі hybrid, про що далі) [47].

Окремо варто виділити ембієнт та атмосферні саундскейпи (комплексні, багатопланові звукові середовища, що створюють атмосферу, настрій і відчуття простору), які часто використовуються як фонові треки в іграх з відкритим світом, жахах або дослідницьких іграх. Такі композиції зазвичай немелодичні або мінімально мелодійні. Їх завдання полягає не в тому, щоб запам'ятися мотивом, а щоб створити певне емоційне середовище для гравця. Ambient-музика використовує тривалі тремтливі звуки, підкладки, реверберації, природні шуми, іноді - легкий гармонічний фон. У ній мелодія, якщо і присутня, то дуже розмита і статична. Як зазначає практикуючий композитор Гаррі Мак, мета ambient-треків - викликати емоцію, не привертаючи уваги до себе [28]. Нерідко протягом десятків секунд можуть звучати просто протяжні акорди або дрони, поступово змінюючись, але без різких подій. Такий звуковий фон дає перепочинок вухам гравця між більш активними музичними моментами і заповнює тишу, яка в іграх небажана [там само].

Атмосферна музика тісно переплітається зі звуковим дизайном. Наприклад, у хоррор-грі замість явної музики можуть звучати атональні ембієнт-шаруваті звуки, такі як гудіння, шепотіння, шорохи, які створюють напруження. В інших випадках межа між музикою і навколишніми звуками розмивається (водночас грає легкий дронний звуковий фон і чути звуки природи). У пригодницьких іграх (квестах, RPG) ambient-треки підкреслюють атмосферу локації: наприклад, у лісі - спокійні етереальні пади з звуками флейти, в космосі - холодні широкі синтезаторні акорди і тихий хор. Ембієнт особливо корисний в ситуаціях, де гравець довго перебуває без зміни діяльності (досліджує місцевість, вирішує загадку), музика тут не повинна набридати. Тому композитори застосовують різні способи варіювання ембієнту: автоматичні фільтри, що час від часу змінюють тембр, випадкові появи тихих мотивів чи звуків, щоб саундтрек залишався живим. Як влучно сказано, іноді гравець «не повинен навіть помічати, що чує якусь музику» небажана, але її відсутність

одразу зробила б сцену порожньою [там само]. Таким чином, ambient-музика - це мистецтво балансу між звуком і тишею, яке формує емоційну канву ігрового світу.

Сьогодні багато ігрових саундтреків тяжіють до змішування жанрів та стилів, утворюючи гібридні форми. Hybrid-музика поєднує, наприклад, оркестрові інструменти з електронними бітом і синтезаторами, або етнічні інструменти з сучасною саунд-дизайнерською обробкою. Метою є створити унікальний саунд для конкретної гри, використовуючи найефективніші елементи різних жанрів. Композитор Мік Гордон характеризував свій підхід до саундтреку «DOOM» не як слідування жанру (метал чи електроніка), а як конструктор із складових: він виділив характерні елементи метал-музики (швидкий темп, важкі гітари, ударні) і окремо елементи електронної індустріальної музики (синтезаторні дрони, спотворені шуми, техно-ритми), а потім комбінував їх, додаючи також хори і струнні для створення власного фірмового звуку [34]. Результат - гібридний стиль, який важко однозначно віднести до якогось одного жанру, але який ідеально підходить атмосфері гри.

Подібні підходи бачимо і в інших проектах: «Halo» знаменито міксує григоріанський хор та електроніку, «NieR: Automata» - камерний оркестр з трип-хоповими ритмами, «The Witcher 3» - слов'янські народні інструменти з епічним оркестром. Гібридність дозволяє підкреслити унікальність ігрового світу: наприклад, для фантастичного сетингу добре підходить поєднання оркестрових підкладок, що асоціюються з кінематографічною епічністю, і синтезаторів, що додають футуристичності. Крім того, комбінування жанрів часто допомагає розкрити різні сторони геймплею: електронний ритм підтримує екшен, а оркестрова мелодика - драматичні сюжетні моменти. З точки зору інтерактивності, гібридні саундтреки можуть надавати і більше можливостей для адаптації. Наприклад, в тихих епізодах може залишатися лише оркестрова складова, а з початком динаміки додаватися електронний біт. Головне при створенні hybrid-музики - досягти органічного злиття стилів. Композитор має глибоко розуміти особливості кожного жанру, щоб їхній союз був

переконливим. Як зазначав Мік Гордон, він спершу аналізує «що робить жанр особливим», бере ці інгредієнти й свідомо експериментує з поєднанням, отримуючи нове звучання [34]. Гібридні стилі стали настільки поширеними в іграх, що нині мало який саундтрек обходиться без елементів змішаних жанрів. Це один із актуальних трендів, що відображає прагнення індустрії до свіжих, несподіваних звукових рішень.

Підсумовуючи, різні музичні стилі у відеоіграх слугують інструментами для досягнення необхідного емоційного ефекту. Оркестрова музика привносить масштабність і глибину, електронна – сучасність і багатство звукових фарб, ambient створює фон і атмосферу, не перевантажуючи слухача, а гібридні стилі дозволяють знайти унікальний тон, який віддзеркалює специфіку конкретної гри. Усі ці стилі можуть існувати в рамках однієї гри (напр., оркестрові теми для головних сцен, ембієнт для дослідження, електроніка для боїв), і завдання композитора – забезпечити їх гармонійне співіснування у динамічній, інтерактивній музичній системі.

1.3. Продакшн адаптивної музики в інтерактивних медіа.

Створення саундтреку для відеогри - це не лише композиція, а й комплексний процес продакшну музики. Сучасний композитор здебільшого виступає одночасно і аранжувальником, і звукорежисером своєї музики. Розглянемо ключові практичні етапи та інструменти, що застосовуються при виробництві інтерактивної музики: використання цифрових студій (DAW) та віртуальних інструментів, особливості запису живих партій, мікшування і фінального мастерингу аудіо перед інтеграцією в гру.

DAW (Digital Audio Workstation) - цифрова аудіо-робоча станція - є центром роботи композитора над ігровою музикою. Популярні DAW (Cubase, Logic Pro, REAPER, Ableton Live та ін.) дозволяють здійснювати запис MIDI-доріжки, програмувати віртуальні інструменти, обробляти аудіо ефектами та

зводити багатоканальні проекти. У контексті інтерактивної музики DAW особливо корисна тим, що дає можливість працювати нелінійно: композитор може створювати декілька варіантів розвитку треку, експериментувати з поєднанням шарів, перевіряти, як будуть звучати перехідні фрагменти. Наприклад, в Ableton Live з його session-view зручно випробовувати різні послідовності кліпів, що імітує горизонтальне ресеквенсування.

Більшість композиторів спочатку повністю реалізують музику в DAW як звичайний трек, а потім експортують необхідні частини: окремі стеми для вертикального методу або сегменти для горизонтального. При роботі над саундтреком великого обсягу проект може містити сотні доріжок, тому вимагається хороша організація: групування треків (струнні, ударні, електроніка тощо), маркування сегментів, використання маркерів для різних станів (наприклад, мітки «Loop A», «Loop B», «Transition»). DAW забезпечує повний контроль над звучанням ще до стадії інтеграції: композитор може почути псевдо-інтерактивну симуляцію, вручну вмикаючи/вимикаючи доріжки і переставляючи блоки, аби переконатися, що задум працює.

Основною «фарбою» в арсеналі ігрового композитора є семплерні бібліотеки та програмні синтезатори. В умовах обмежених бюджетів та строків часто неможливо записати живий оркестр для кожного треку, тому використовуються високоякісні записи інструментів (семпли), що відтворюються через комп'ютер. Такі бібліотеки (напр. Spitfire Albion, EastWest Hollywood Strings, Metropolis Ark тощо) дозволяють отримати реалістичне звучання струнних, духових, ударних і хору, не виходячи зі студії. Семплові бібліотеки стали хлібом насущним композиторів, фактично обов'язковим інструментом при створенні ігрової музики. Відомо, що навіть у топових проєктах розробники часто віддають перевагу хорошим семплам через економію - наприклад, багато успішних ігор мають саундтреки, зроблені повністю «in the box» (тобто без живих записів), використовуючи лише електронні бібліотеки, і пересічний гравець цього не помічає.

Композитор, плануючи саундтрек, підбирає необхідні бібліотеки під стилістику: оркестрові (струнні, мідь) - для епічної музики, аналогові синтезатори - для ретровейв атмосфери, етнічні інструменти - для автентичності сетингу тощо. На практиці часто доводиться поєднувати кілька бібліотек, щоб досягти бажаного результату. Кожна бібліотека має свої сильні сторони: одна краще передає легато струнних, інша - акцентовані стакато, і композитор може використовувати їх разом, компенсуючи недоліки одна одної. Це вимагає і технічної майстерності (узгодити різні просторові характеристики семплів, налаштувати еквалізацію, щоб інструменти з різних бібліотек злилися). Для зручності часто обирають бібліотеки, записані в схожих умовах - наприклад, всі з «сухим» студійним звуком, щоб потім додати свою реверберацію і створити єдине акустичне середовище.

Окрім оркестрових семплів, багато ігор використовують синтезатори. Програмні синтезатори (Serum, Massive, Omnisphere та ін.) або модульні аналогові синтезатори дають можливість створювати нові тембри. Композитори нерідко витрачають час на саунд-дизайн - конструювання власних звуків, щоб саундтрек був унікальним. Наприклад, виття монстрів може бути зроблено через синтез генератором шуму, а магічний ефект - через гранулярний синтез падів. Рекомендовано уникати шаблонних пресетів і створювати свої патчі, адже використання готових пресетів загрожує тим, що музика стане схожою на інші роботи. У підсумку, віртуальні інструменти дають композитору оркестр під рукою, хор, будь-який етнічний інструмент та необмежений світ електронних звуків. Важливо вміти цим різноманіттям керувати, щоб не потонути у можливостях і водночас отримати оригінальний саундтрек.

Та попри досягнення технологій, жоден семплер не передасть на 100% виразність живого виконання. Тому у продакшні ігор часто комбінують живі і vst інструменти: ключові партії (соло-інструменти, вокальні теми, перкусія) записуються з реальними музикантами, а решта доробляється бібліотеками. Залежить від бюджету: інді-розробники можуть дозволити собі, скажімо, сесійного гітариста для запису головної гітари, або сольного скрипаля. ААА-

проекти - це вже масштабні сесії з оркестром і хором. Процес запису для інтерактивної музики має свої нюанси: потрібно записати не цілі твори, а фрагменти і шари окремо. Оркестр можуть записувати по секціях, щоб мати контроль над кожним шаром у міксі гри. Також запис проводять з клік-треком на фіксованому темпі, щоб потім аудіо можна було ідеально лупувати чи синхронізувати при перемиканнях. Іноді окремо записують різні варіанти інтенсивності виконання (наприклад, піано і форте версії партії струнних), які потім використовуються як варіанти вертикальних шарів. Живий вокал - популярний прийом для надання саундтреку емоційності; у інтерактивній музиці вокал часто застосовують тематично (лейтмотив персонажу або культова пісня у кінці гри). У клік метронома записують можливі і варіативні фрази, щоб при повторенні не було монотонності. Після запису настає етап редагування: аудіо порційно вирізають, підганяють під потрібну довжину лупів, чистять від шумів. Результатом цього етапу є набір аудіофайлів, стемів або сегментів, готових до мікшування.

Мікшування ігрової музики - складний, багаторівневий процес. По-перше, звукорежисер зводить музику як окремий твір: баланс між інструментами, панорамування, еквалізація, реверберація (всі стандартні студійні процедури для досягнення якісного саундтреку). Тут є одна особливість: якщо передбачається динамічне мікшування в грі, тобто коли гра сама змінює гучність шарів чи доріжок під час виконання (наприклад, заглушує музичний фон під час діалогу або робить затухання при паузі), то на стадії зведення треба забезпечити запас динаміки. Мікс не повинен бути занадто стислим чи гучним, навпаки, краще залишити простір, адже в грі часто використовується стандарт -23 LUFS (рівень гучності, прийнятий для інтеграції аудіо у медіа). Тому саундтрек для гри зводять з більшою динамікою, ніж, скажімо, поп-музику, щоб не виникало спотворень при накладенні звукових ефектів або діалогів.

По-друге, потрібно врахувати сумісність з ігровим звуковим середовищем. Мікс музики коригується з урахуванням того, що паралельно звучатимуть шуми, діалоги тощо. Зазвичай частоти низького і середнього діапазону (голос, вибухи)

резервують для звукових ефектів, а музику трохи «відсовують» еквалайзером (наприклад, трохи знижують рівень у діапазоні 500 Гц - 2 кГц, щоб не заважати важливим ігровим звукам). Так само, якщо у грі постійно звучить шумовий фон (вітер, дощ), музичний мікс можна зробити більш яскравим у верхній середині, щоб не тонував у тих шумах. Зведення інтерактивного саундтреку також включає підготовку альтернативних міксів: звукорежисер може зробити окремо мікс лише ритмічної секції і лише мелодичної (для різних станів). Звукові рушії (FMOD/Wwise) дозволяють в реальному часі керувати міксом, тож завдання музичного продюсера - надати оптимально збалансовані стери. Наприклад, у вертикальній системі кожен шар повинен сам по собі звучати повноцінно (не бути надто «голим» або перенасиченим). Тому іноді до складу кожного шару додають трохи спільного ревербераційного простору або фонового шуму, щоб при соло-програванні шар не був відверто ізольованим від інших. По суті, міксування для інтерактивної музики - це міксування у кількох вимірах: по горизонталі (між послідовними фрагментами) і по вертикалі (між одночасними шарами).

Завершальна обробка музичних треків перед впровадженням у гру - це мастеринг. Він відрізняється від класичного альбомного мастерингу. Якщо в альбомі усі треки призводять до єдиного стандартного рівня гучності і тонального балансу, то в грі різні треки можуть призначатися для різних ситуацій (спокійний бекграунд vs. бойова тема) і не потребують однакової гучності. Натомість мастеринг зосереджується на технічній якості: прибрати кліпи, виставити правильний рівень сигналу для ігрового движка (зазвичай пік рівня встановлюють трохи нижче 0 dBFS, наприклад -1 або -2 dB, щоб уникнути цифрового перевантаження при сумуванні з іншими звуками). Також застосовується легка компресія і лімітинг для контролю динамічного діапазону: в грі занадто широкий діапазон може означати, що тихі частини загубляться за звуковими ефектами. Але перекачувати музику (робити її ультра-гучною) теж не можна: як згадано, часто орієнтуються на -23 LUFS інтегрованої гучності для музичного міксу. Мастеринг іноді включає стерео-поліпшення чи еквалізацію,

якщо потрібно підкоригувати загальне звучання під гру (скажімо, додати низьких частот, якщо звуковий рушій їх обрізає, або трохи підняти «повітря» на 10-12 кГц для прозорості).

Ще одна задача мастерингу - конвертація у потрібний формат. Музику часто зберігають у стисненому вигляді (Ogg Vorbis, AAC тощо) для економії місця, але компресія не має помітно погіршити якість. Тому мастер-трек може бути 24-bit WAV, а вже ігрові файли - Vorbis ~192-320 kbps. Під час мастерингу перевіряють, як трек звучить після компресії, чи нема артефактів. Крім того, якщо в проєкті є адаптивні переходи, майстеринг охоплює перевірку швів: кінцівки сегментів повинні правильно затухати (або мати спеціальний хвіст реверберації, який не обрізається при лупі), рівні суміжних фрагментів – узгоджені, щоб не було стрибка гучності.

У підсумку, продакшн інтерактивної музики - це синтез творчості і техніки. Композитор повинен бути обізнаний зі студійними інструментами на рівні звукорежисера. Він відкриває DAW, завантажує улюблені бібліотеки і починає творити, тримаючи в голові, як усе це потім розпадеться на шматочки у грі. Необхідно володіти навичками міксування багатоканальних шарів, вміти записати за потреби «живий» звук, і все це - з урахуванням подальшої інтеграції. Правильно вибудований процес продакшну забезпечує, що фінальний саундтрек звучатиме якісно та ефектно в умовах реального геймплею, де гучні вибухи можуть раптово заглушити скрипку, а повторений уп'яте луп все ще має викликати емоцію. Саме на етапі продакшну музика набуває того вигляду, в якому її почує гравець, тому цей етап не менш важливий, ніж написання нот.

Після того, як музичні файли підготовлені, постає завдання інтегрувати їх у гру так, щоб реалізувати всю задуману адаптивність. Тут на допомогу приходять спеціалізовані аудіо-середовища - middleware-платформи (проміжне програмне забезпечення для аудіо). Найбільш відомі рішення - це Audiokinetic Wwise та Firelight FMOD, які стали індустрійним стандартом інтеграції звуку та музики до ігор. Middleware слугує прошарком між ігровим рушієм (Unity, Unreal Engine тощо) і безпосередньо звуковими ресурсами, надаючи зручні інструменти

для побудови інтерактивної звукової системи без глибокого занурення в програмування аудіо. Але повернемося до цього в наступному розділі.

Теоретичні принципи інтерактивної музики найкраще ілюструються реальними прикладами з ігор, де ті чи інші методи були успішно втілені на практиці. Розглянемо декілька знакових саундтреків та підходів їхніх авторів - зокрема, досвід композиторів Остіна Вінторі, Гарета Кокера, Міка Гордона та інших, чії роботи стали еталонами в сфері музики для відеоігор.

«Journey» (2012), де композитором виступив Остін Вінторі, увійшла в історію як гра з надзвичайно цілісним і емоційним музичним супроводом, який фактично розповідає історію без слів. Вінторі застосував у «Journey» вертикальний леєринг в поєднанні з тонко продуманою тематичною роботою. Саундтрек складається з кількох основних тем, що відповідають різним етапам подорожі, але всередині кожної з них музика розвивається шляхом додавання або прибирання шарів. Наприклад, у грі є момент, коли гравець рухається пустелею: спершу лунає лише соло віолончелі (головна тема), згодом, коли відкривається ширший ландшафт - додається хор, струнні, розгортаючи емоційне полотно. Якщо гравець зупиняється або діє повільно - аранжування залишається тонким, лише кілька інструментів; якщо просувається швидко, досягає певної точки - вступає повне оркестрове наповнення. Ця плавність досягається через те, що всі музичні шари написані так, щоб гармонійно поєднуватися у будь-якій комбінації, і гра непомітно міксує їх залежно від прогресу. Як зазначалося самим композитором, він прагнув створити відчуття, що музика «дихає разом із гравцем» [18]. Вертикальний метод у «Journey» реалізовано блискуче: шари змінюються не різко, а поступово, через динамічні відтінки та підтримуючі оркестрові голоси, тому гравець часто навіть не усвідомлює, що музика адаптивна, вона здається єдиним потоком, який просто природно наростає або стихає. «Journey» стала одним з перших інді-саундтреків, номінованих на премію Греммі, і це багато в чому завдяки інтерактивному музичному досвіду, що створив Вінторі. Його підхід надихнув багатьох композиторів використовувати невеликі ансамблі й солюючі інструменти у

поєднанні з шаруватою електронікою для досягнення глибокого емоційного резонансу при мінімалізмі засобів.

Гарет Кокер відомий своїм мелодизмом та оркестровим розмахом у іграх серії «Ori». Його саундтреки вирізняються багатою тематичною палітрою та використанням етнічних інструментів поряд із оркестром. В плані інтерактивності ігри «Ori» використовують переважно горизонтальну сегментацію музики з окремими треками для різних ситуацій (мирні зони, небезпека, погоні тощо). Цікаво, що Кокер в інтерв'ю відзначав: вони експериментували з частим переключенням музики на дрібні події (на кшталт появи дрібних ворогів), але виявили, що надто часті зміни музичної теми звучать «занадто штучно, по-ігровому клішовано» і можуть дратувати [1]. Тому було прийнято рішення не змінювати музичну тему при кожній сутичці з дрібним противником. Замість цього музика змінюється тільки в ключові моменти або в більших битвах.

Наприклад, під час дослідження локації музика ллється ніжно і безперервно, навіть якщо гравець вступає в короткі бої - це не приводить одразу до перемикання на «бойову тему». Лише якщо герой потрапляє в тривалу напружену секцію або бос-битву, саундтрек переходить на інший рівень інтенсивності. Кокер рекомендує шукати в грі «тригери» - важливі точки сюжету чи геймплею, де зміна музики буде найбільш доречною і драматично виправданою. Такий підхід можна назвати кінематографічним: музика пишеться і інтегрується так, ніби супроводжує історію, де є ключові сцени, і між ними вона не «смикається» через дрібниці. Це підвищує цілісність сприйняття.

Водночас в «Ori» застосовано і вертикальні елементи - наприклад, у великих битвах до основного треку можуть додаватися хорові або ударні шари, якщо сутичка затягується або стає більш небезпечною (тобто поєднується горизонтальний вибір теми і вертикальне нарощування її потужності). Крім того, Ori демонструє майстерне використання лейтмотивів: Кокер не соромиться повторно вводити теми в різних ітераціях. У «Will of the Wisps» він навіть повертає основну тему з першої частини у кульмінаційні моменти другої, щоб

викликати у гравців ностальгійні емоції. Це показує, що інтерактивність не заперечує класичних композиційних технік розвитку тематичного матеріалу. Навпаки, грамотне поєднання лейтмотивів із динамічною структурою підсилює емоційний зв'язок гравця з музикою. Саундтреки «Ori» здобули численні нагороди, а сам Гарет Кокер підкреслює, що ключ до успіху - тісна співпраця з командою розробників і глибоке розуміння ігрового процесу, адже тільки так можна знайти ідеальні точки і способи адаптації музики під гру [1].

Коли id Software перезапускали легендарну «DOOM», перед композитором Міком Гордоном стояло завдання: створити сучасний саундтрек, який зберіг би дух оригіналу (метал, агресія), але водночас здивував би гравців. Цікаво, що однією з умов від розробників було «жодного прямолінійного металу» - вони хотіли інноваційний підхід [34]. Гордон знайшов рішення у гібридному стилі: він змішав елементи важкого металу (електрогітари з екстремальним дісторшном, швидкі ударні) з дарк-електронікою (потужні синтезаторні басы, noise-ефекти) та індастріалом.

З точки зору інтерактивності, музика «DOOM» реалізована через систему шарів інтенсивності. У спокійні моменти грає лише відносно тихий ембієнт-трек з віддаленими звуками (часто це розтягнуті гітарні дрони). Щойно починається бій, до нього додаються нові елементи: ритмічний барабанний грук, потім важкий гітарний риф. Якщо бій загострюється (багато ворогів, гравець у небезпеці) - підключаються максимальні шари: партії для бас-барабана, зіграні на подвійній педалі (кардані), агресивні синтезаторні арпеджіо, додаткові гітарні партії. Ця вертикальна ескалація зроблена так, що музика завжди синхронна по темпу, і шар нарощується на плавному фейді, тому гравець чує просто «музика стала більш лютою», не відчуваючи, що це окремий трек. Головна тема «BFG Division» фактично розбита на кілька компонентів, які мікшуються в реальному часі.

Мік Гордон в інтерв'ю казав, що сприймає музику гри як частину відчуття геймплею: музика є винагородою для гравця [34]. У «DOOM» це відчувається буквально. Коли починає грати «крутий» трек, значить ви все робите правильно,

і саундтрек вас підбадьорює. У технічному плані, саундтрек «DOOM» (2016) часто приводять як зразковий кейс адаптивної музики в екшнах. Він демонструє, як швидкі вертикальні зміни можуть бути ефективними: гравець постійно в динаміці бою, відповідно музика то затихає (між хвилями ворогів), то розганяється до максимуму, і це відбувається невимушено, бо Гордон написав треки спеціально модульними, передбачивши їх таку поведінку. Саундтрек отримав численні нагороди «Найкращий саундтрек» у ігровій індустрії, а Мік Гордон став одним з найвідоміших композиторів нового покоління, підхід якого тепер наслідують у багатьох екшен-іграх, що бажають поєднати сучасний електронний саунд дизайн з традиціями ігрового металу.

Наведені приклади підкреслюють, що інтерактивна музика - це не вузький набір прийомів, а цілий спектр творчих стратегій. Кожен композитор знаходить власне рішення, виходячи з потреб гри. Остін Вінторі показав силу тонкої емпатичної музики, що плавно реагує на гравця; Гарет Кокер - важливість драматургічного підходу і використання тригерів замість дрібних реакцій; Мік Гордон - енергію і драйв гібридного, шарами нарощуваного саундтреку як частину геймплею. Всі вони опираються на описані теоретичні принципи, але трансформують їх у свій унікальний стиль. Таким чином, теорія аранжування та продакшну в інтерактивних медіа продовжує розвиватися завдяки новаторству композиторів-практиків, що розширюють горизонти того, якою може бути музика в інтерактивному середовищі.

РОЗДІЛ 2. ТЕХНОЛОГІЧНІ АСПЕКТИ ІМПЛЕМЕНТАЦІЇ АДАПТИВНОЇ МУЗИКИ В ІГРОВИХ РУШІЯХ

2.1. Огляд сучасних ігрових рушіїв та middleware (на прикладі Unity/Unreal Engine, Wwise/FMOD)

У попередніх розділах було окреслено композиторсько-технологічні засади адаптивного аранжування: від типології інтерактивності (вертикальне нашарування, горизонтальне ресеквенування, параметричні та стано-керовані моделі) до принципів синхронізації музичних переходів і вимог до контенту. Наступним логічним кроком є визначення, на яких інструментах і як саме ці підходи втілюються у реальному виробничому циклі гри.

Цей розділ виконує роль технологічного мосту між композиторським задумом і ігровою реалізацією. Ми розглянемо, які можливості надають провідні ігрові рушії Unity та Unreal Engine для імплементації інтерактивного аудіо на рівні/сцені, подій та ігрових станів; а також проаналізуємо спеціалізовані аудіоплатформи Wwise та FMOD, що забезпечують проектування й керування адаптивною музикою в авторинговому середовищі з подальшою інтеграцією в рушій.

Фокус зроблено на тих характеристиках, які безпосередньо впливають на музичну драматургію: підтримка темпоритмічної сітки та музичних маркерів, механіки плавних переходів і кросфейдів, робота з шарами (стемами), RTPC/параметричне керування, станова логіка, просторове мікшування, профайлінг і оптимізація під цільові платформи. Для кожного інструмента надамо коротку архітектурну довідку, типові сценарії застосування та приклади проектних рішень.

Ігровий рушій Unity забезпечує гнучку базову аудіосистему, орієнтовану на простоту використання. Звуки прив'язуються до об'єктів через аудіокомпоненти (Audio Source), а прослуховування здійснюється через компонент Audio Listener на камері [50]. Такий компонентно-орієнтований

підхід полегшує початкову роботу зі звуком для розробників. Unity підтримує прості форми просторового звуку, реверберації та інших ефектів за допомогою аудіомікшера і скриптів. Однак для реалізації адаптивної музики (динамічного змінення музичного супроводу під час гри) штатних засобів Unity часто недостатньо. Рушій не має вбудованої високорівневої системи інтерактивної музики, тому складні сценарії – наприклад, плавне перемикання музичних тем або шарів в залежності від дій гравця потребують додаткового програмування або залучення зовнішніх аудіомодулів [50]. В Unity можливо створювати адаптивну музику шляхом скриптового керування гучністю та параметрами аудіомікшера, проте такий підхід вимагає більше зусиль від розробників і не настільки зручний, як використання спеціалізованого middleware.

Unreal Engine відомий потужною вбудованою аудіопідсистемою та інструментами для складного звукового дизайну. В Unreal використовується вузловий підхід (node-based): звукові події та ефекти налаштовуються через звукові графи Sound Cue, де різні вузли визначають поведінку звуку (відтворення, затухання, випадкові варіації тощо) [50]. Також Unreal надає Blueprint - візуальну скриптову систему, що дозволяє налаштовувати аудіологіку без написання коду. Це дає саунд дизайнерам можливість створювати інтерактивні аудіосцени, прив'язуючи зміни музики чи звуку до подій гри на рівні блок-схем [Ошибка! Источник ссылки не найден.]. Unreal Engine 4 і 5 мають розширені можливості обробки звуку в реальному часі: просторове звучання, реверберацію, оклюзію, мікшування з використанням Sound Class та Sound Mix, а у версії UE5 запроваджено систему MetaSounds для процедурної генерації звуку. Все це дозволяє Unreal реалізувати адаптивне аудіо навіть без сторонніх бібліотек. Зокрема, адаптивна музика може бути втілена засобами Blueprint: рушій дозволяє змінювати темп, інтенсивність або композицію музики залежно від стану геймплею [Ошибка! Источник ссылки не найден.]. Наприклад, при появі ворогів можна по сценарію Blueprint збільшити гучність або додати нові музичні шари для посилення напруги, а після битви - плавно знизити інтенсивність музики. Таким чином, Unreal Engine із коробки пропонує

багатші засоби для інтерактивного аудіо, ніж Unity [50], що зменшує потребу в додатковому middleware для простих і середніх за складністю аудіозадач.

Обидва рушії підтримують інтеграцію зовнішніх аудіосистем і надають базові можливості для роботи зі звуком, проте є суттєві відмінності у підходах. Unity простіший у засвоєнні для початківців завдяки компонентній моделі, але реалізація складної адаптивної музики в ньому потребує більше ручної роботи або плагінів [50]. Unreal Engine, завдяки вузловій системі та Blueprints, забезпечує більш візуальний і потужний інструментарій для аудіодизайну «з коробки» [Ошибка! Источник ссылки не найден.]. Наприклад, типова задача (зробити так, щоб музика в грі змінювалася від спокійної до напруженої при вступі в бій) в Unreal може бути вирішена засобами Blueprints з використанням готових вузлів і параметрів, тоді як в Unity вимагатиме написання C#-коду або використання middleware. Крім того, Unreal має більше вбудованих ефектів та можливостей просторового звуку без додаткових плагінів. Водночас, гнучкість Unity проявляється у можливості розширення: існують офіційні інтеграції для Wwise та FMOD, що дозволяють долучити їхні можливості до проекту Unity практично на рівних з Unreal умовах [50]. Отже, вибір рушія для аудіо часто залежить від масштабу та потреб проекту: для великих ігор з високими вимогами до звуку нерідко обирають Unreal Engine за його багатий аудіоарсенал, тоді як Unity є популярним серед інди-розробників, які можуть підключити middleware за потреби для розширення аудіоможливостей.

Аудіо middleware - це проміжне програмне забезпечення, яке інтегрується між ігровим рушієм та аудіоконтентом. Його основна мета - надати звукорежисерам та композиторам спеціалізовані інструменти для створення інтерактивного звуку та музики без необхідності глибокого програмування в рушії [50]. Використання middleware розв'язує низку проблем, з якими стикаються розробники, якщо покладатися лише на вбудовані аудіосистеми рушіїв. По-перше, воно забезпечує покращений контроль та якість звуку: професійні бібліотеки ефектів, гнучкі налаштування 3D-звучання, об'ємного аудіо, оклюзії тощо, які важко реалізувати силами стандартних засобів. По-

друге, middleware значно спрощує робочий процес: звукові дизайнери можуть самостійно вносити зміни і прослуховувати результати в режимі реального часу, не залучаючи постійно програмістів [54]. Зокрема, системи на кшталт Wwise та FMOD підтримують live update (можливість підключитись до запущеної гри і відразу коригувати мікс чи параметри звуку). По-третє, аудіо-middleware проєктувалося з розрахунком на адаптивність та крос-платформеність: такі рушії полегшують створення динамічного звукового середовища, яке реагує на події гри, і гарантують однакове звучання на різних платформах [там само]. Для сучасних ігор, особливо масштабних, це критично. Музика та звукові ефекти повинні невимушено підлаштовуватися під дії гравця, посилюючи ефект присутності.

Найпоширенішими рішеннями аудіо-middleware у індустрії є Audiokinetic Wwise та FMOD Studio. Обидва вони добре інтегруються з популярними рушіями (Unity, Unreal та ін.) і надають можливості, яких бракує вбудованим системам, зокрема: адаптивна музика, інтерактивні звукові ефекти, складні схеми мікшування, управління аудіопотоками тощо [54; 50]. Нижче розглянемо кожен із них детальніше, а також їхні типові можливості для реалізації інтерактивної музики.

Audiokinetic Wwise (Wave Works Interactive Sound Engine) - потужний аудіорушій, який широко застосовується у AAA-іграх. Wwise надає розробникам окреме середовище програмування звуку, в якому можна централізовано керувати всіма аспектами аудіо: імпортувати та організувати файли, застосовувати ефекти, визначати параметри та сценарії відтворення звуку [55]. Однією з ключових особливостей Wwise є спеціальна система Interactive Music - підсистема, призначена для реалізації адаптивних музичних партитур. Вона дозволяє розбити саундтрек на музичні фрагменти (сегменти) і задавати правила їхнього чергування або накладання залежно від перебігу гри. Наприклад, композитор може підготувати декілька варіантів музичної теми різної інтенсивності (спокійна, напружена, бойова), а в Wwise налаштувати перехід між ними за допомогою станів гри або параметрів (RTPC - Real-Time Parameter

Controls). Це забезпечує горизонтальну адаптацію музики (зміну фрагментів у часі) та вертикальну (нашарування додаткових музичних доріжок). Обидві техніки підтримуються інструментарієм Wwise [54].

Wwise пропонує зручні засоби для реалізації вертикального мікшування: композитор може завантажити в проєкт окремі теми (шари) музики (наприклад, ударні, струнні, хор) і керувати їхньою активністю. У відповіді на події гри Wwise може автоматично додавати або прибирати певні шари, змінюючи таким чином настрій музики динамічно [54]. Так само гнучко налаштовуються горизонтальні переходи: у середовищі Wwise задаються точки переходу між музичними сегментами, умови (тригери) для цих переходів та типи згасання/наростання звуку при зміні теми. Важливо, що всі ці складні сценарії реалізуються засобами самого Wwise, без потреби писати код у грі, достатньо один раз інтегрувати Wwise в рушій і викликати події (Events) Wwise з коду гри чи із Blueprints.

Серед відомих ігор, де Wwise використовувався для музики, є серія «Assassin's Creed». Звукорежисери Ubisoft побудували на Wwise інтерактивну музику, яка змінюється залежно від регіону, карти чи рівня загрози: спокійна мелодія під час дослідження світу плавно переходить у напружений трек при переслідуванні ворогів [54]. Інший приклад - «Cyberpunk 2077», де Wwise забезпечив складний звуковий ландшафт мегаполісу і динамічний саундтрек, що реагує на дії гравця (бойові сцени супроводжуються підсиленням ритмічних електронних мотивів) [55]. Wwise славиться масштабованістю: він однаково добре працює як з нечисленними звуками інді-гри, так і з тисячами аудіооб'єктів великого проєкту. Це підтверджується переліком проєктів: Wwise інтегровано в движки Unreal Engine 3/4/5, Unity, CryEngine та багато інших комерційних рушіїв, а список ігор, що його використовують, нараховує сотні найменувань: від мобільних казуальних до топових AAA-блокбастерів.

Перевагами Wwise є надзвичайно багатий функціонал для аудіо (велика бібліотека ефектів, підтримка просторого звуку й стандартів Dolby Atmos, багатоплатформеність), гнучка система інтерактивної музики, інструменти

профайлінгу та оптимізації аудіо. Для великих команд важливо, що Wwise дозволяє декільком звукорежисерам одночасно працювати над проектом та має засоби контролю версій. До недоліків можна віднести відносно високу складність освоєння. Також комерційна ліцензія Wwise для великих проектів платна, що потребує врахування в бюджеті (для інді-ігор з невеликим доходом Audiokinetic надає безкоштовну чи недорогу ліцензію) [55].

FMOD Studio - інша провідна аудіоплатформа, яка особливо популярна серед незалежних розробників та середніх студій. FMOD, як і Wwise, пропонує окремий інтерфейс для дизайну звуку і музики, інтегрований з ігровим рушієм через API або плагіни. Головний акцент FMOD - простота та інтуїтивність: він має drag-and-drop інтерфейс, схожий на цифрову аудіостанцію (DAW), що дозволяє швидко створювати події звуку, міксувати їх та призначати параметри без написання коду [54]. Це робить FMOD привабливим вибором для менших команд, де саунд-дизайнер може не мати підтримки окремого програміста звуку.

Для реалізації адаптивної музики FMOD надає концепцію музичних подій (Events) з таймлайном. Композитор може завантажити декілька музичних доріжок на таймлайн FMOD і задати маркери переходів, параметри та логіку відтворення. Наприклад, можна створити подію «Бойова музика», яка містить два розділи: спокійний вступ і інтенсивну основну частину. За допомогою параметра (напр. «інтенсивність бою» від 0 до 1) в FMOD налаштовується, що при значеннях нижче 0.5 грає перший розділ, а коли параметр перевищує 0.5, відбувається перехід на другий розділ з плавним переходом (cross-fade). Така параметрична система дуже гнучка: параметри можна прив'язувати до будь-яких ігрових метрик (здоров'я гравця, кількість ворогів, фаза рівня тощо), і музика буде в реальному часі підлаштовуватися під зміну цих значень [50]. FMOD також підтримує вертикальне нашарування: на одній події можна розмістити кілька доріжок, які представляють різні інструментальні шари, та керувати їх гучністю через параметри або Automations (вбудовані криві зміни параметрів).

Характерний приклад - саундтрек інді-гри «Celeste». У «Celeste» композитор Лена Рейн створила адаптивну музику, де з проходженням рівня

додаються нові музичні шари. Це було реалізовано у FMOD: кожен рівень мав базову мелодію, а при досягненні певних точок гравцем FMOD додавав додаткові інструментальні треки, збагачуючи аранжування. Інша відома гра - «Hades» від Supergiant Games - використовує FMOD Studio для побудови динамічного саундтреку: спокійна музика грає під час дослідження підземелля, але як тільки починається бій, FMOD перемикається на агресивнішу частину треку і вмикає додаткові «рок»-шари; після завершення бою система точно в такт завершує музичну фразу, щоб підкреслити перемогу гравця [31]. Ця плавність переходів була досягнута саме завдяки можливостям FMOD синхронізувати музичні сегменти з подіями гри.

Перевагою FMOD є нижчий поріг входження (багато хто відзначає, що освоїти FMOD легше, інтерфейс зрозумілий для людей, знайомих із аудіоредакторами. FMOD має безкоштовну ліцензію для проєктів з невеликим бюджетом, що робить його доступним для інди-розробників. Незважаючи на простоту, FMOD підтримує всі ключові функції: 3D-звук, DSP-ефекти, роботу з параметрами, живе оновлення звуку в грі. Недоліком можна вважати те, що для надзвичайно масштабних і складних аудіопроектів FMOD може поступатися Wwise у деяких просунутих можливостях і оптимізації (наприклад, у Wwise більш розвинена система керування ресурсами і пам'яттю для дуже великих ігор) [54]. Втім, багато успішних проєктів різного розміру працюють на FMOD, від невеликих інди до АА-проектів, що свідчить про його універсальність.

Обидва middleware-рішення покривають схожий набір задач, однак між ними є відмінності, що впливають на вибір інструменту для конкретного проєкту. Wwise традиційно вважається вибором для великих студій та ААА-ігор, де потрібен максимально багатий функціонал і тонкий контроль. Відомі серії на кшталт «Assassin's Creed», «Destiny», «Borderlands» використовують Wwise для створення комплексних звукових ландшафтів. FMOD, своєю чергою, дуже популярний серед незалежних розробників і невеликих команд завдяки простоті та лояльній ліцензійній політиці. Приміром, «Celeste», «Hollow Knight», «Undertale» та десятки інших інди-ігор покладаються на FMOD для інтерактивної

музики і звукових ефектів. З огляду на це, можна умовно сказати: для масштабних проєктів з великою кількістю аудіоактивів та складною логікою більше підходить Wwise, тоді як для швидкого прототипування і менших ігор FMOD може бути оптимальним вибором [54].

З точки зору функціональності, Wwise і FMOD багато в чому рівноцінні: обидва підтримують вертикальне нашарування музики, горизонтальне перекомпонування, реакцію на параметри гри в реальному часі, 3D-аудіо тощо. Wwise дає трохи більше можливостей для кастомізації та має багатший інструментарій із коробки (наприклад, готові рішення для управління складними ієрархіями об'єктів, вбудовані ефекти високого класу). FMOD вирізняється більш зручним UI і швидким робочим процесом: багато дій виконуються шляхом перетягування, налаштування переходів і параметрів візуально на таймлайні, що часто прискорює роботу для невеликих команд.

Варто зазначити, що і Wwise, і FMOD мають офіційні інтеграції практично з усіма сучасними рушіями. Для Unity та Unreal Engine Audiokinetic і Firelight Technologies (розробник FMOD) підтримують готові плагіни, що забезпечують «безшовне» підключення: події звуку, створені в цих middleware, можна викликати прямо з коду Unity (C#) або з Blueprint/C++ в Unreal [50]. Це означає, що вибір Wwise vs FMOD не обмежує вибір рушія - обидва працюватимуть і в Unity, і в Unreal, різниця лише в деталях реалізації. Таким чином, розробники можуть обирати middleware, відштовхуючись від потреб проєкту та компетенції команди, не боячись проблем сумісності.

В Unity інтеграція FMOD, наприклад, дозволяє бачити студійні події FMOD як об'єкти в Unity: розробник може розмістити на сцені префаб «Studio Event Emitter» і вказати, яку саме подію відтворювати, а далі керувати нею через компоненти або код [54]. Схожим чином працює інтеграція Wwise: вона додає в Unity власний менеджер аудіосцен (Wwise Global), а також компоненти типу «Ak Audio Listener», «Ak Ambient» тощо, що представляють собою аналоги Unity Audio Source, але керовані Wwise. Завдяки цьому звукорежисер може налаштовувати більшість поведінки звуку у Wwise Authoring, а програміст - лише

запускати події в кодї C# для програвання. Це значно прискорює розробку і мінімізує кількість коду, пов'язаного зі звуком.

В Unreal Engine інтеграція ще глибша: і Wwise, і FMOD надають можливість використовувати свої події безпосередньо в Blueprints. Для Wwise існує набір Blueprint-нодів (AkAudio Events), які дозволяють, наприклад, постити подію, встановлювати значення RTPC-параметра, керувати станами (States) тощо без написання коду на C++ взагалі. FMOD аналогічно пропонує Blueprint nodes, як-» «Play Event at Location», «Set Parameter» і т.д. Так, дизайнер рівнів в Unreal може візуально розмістити тригер, що при вході гравця в зону викличе подію зміни музики FMOD, і налаштувати це кількома кліками. Подібна гнучкість особливо корисна при роботі з адаптивною музикою, бо можна швидко налаштовувати точки переключення музичних тем, перевіряти їх прямо в редакторі. В результаті Unreal + middleware дає максимум свободи аудіокоманді: внутрішні засоби рушія (наприклад, Blueprints, матриця мікшування звуку) комбінуються з можливостями middleware. Наприклад, гру «Gears 5» (розроблену на Unreal Engine 4) озвучено за допомогою Wwise: розробники використали Blueprints для відстеження ігрових подій та передачі їх у Wwise як параметрів, що негайно відбивалося на музичній темі або на звукових ефектах оточення [55].

На практиці сьогодні важко знайти велику гру, де інтерактивний звук реалізовано без використання стороннього аудіо-движка. Ось декілька показових прикладів застосування Unity/Unreal з Wwise/FMOD у реальних проектах:

- «Firewatch» (2016) - пригодницька гра від Campo Santo на рушії Unity. Для озвучення було інтегровано Wwise, що дозволило створити динамічну атмосферну звукову картину лісу і адаптивний саундтрек. Зокрема, музика в Firewatch змінюється залежно від прогресу в сюжеті та місцезнаходження гравця: Wwise керує плавними переходами між спокійними мелодіями дослідження та тривожними темами у напружені моменти. Доповідь на GDC 2016 від розробників

Firewatch підтвердила, що використання Wwise у поєднанні з Unity значно спростило імплементацію всіх звукових систем гри [50].

- «Gears 5» (2019) - шутер від третьої особи від студії The Coalition, розроблений на Unreal Engine 4. Гра використовує Wwise як аудіомідлвар: це дозволило реалізувати комплексний інтерактивний саундтрек для кампанії. Наприклад, в залежності від інтенсивності бою Wwise додає або відсікає певні музичні шари, а спеціальні музичні стінгери (короткі фрагменти) автоматично запускаються при ключових подіях (як-от перемога над босом), посилюючи емоційний вплив моменту. В рушії Unreal розробники налаштували Blueprints, що відправляють відповідні повідомлення у Wwise - останній вже виконує заздалегідь підготовлений композитором сценарій перемикання музики. В результаті гравці відзначали високий рівень занурення завдяки музиці, яка «відчуває» геймплей.

- «Celeste» (2018) - інді-платформер (рушій Monogame), відомий своєю адаптивною музикою. Хоча гра не на Unity чи Unreal, вона демонструє можливості FMOD: композиторка Лена Рейн використала FMOD Studio для того, щоб кожен рівень мав кілька варіацій треку. Коли гравець просувається вперед і напруга зростає, FMOD поступово додає нові шари (ритмічний бас, перкусія тощо) до музики, створюючи відчуття розвитку композиції разом із прогресом гравця. Якщо ж гравець гине і починає спочатку секцію - музика плавно повертається до початкового спокійного стану. Цей підхід (поєднання вертикального нашарування і горизонтальних переходів) був легко реалізований через події і параметри FMOD. «Celeste» стала показовим кейсом для спільноти інді-розробників щодо використання middleware: навіть без великого бюджету, завдяки FMOD, невелика команда досягла AAA-якості інтерактивного звуку.

- «Hades» (2020) - популярна roguelike-гра від Supergiant Games. Розробники використали FMOD для створення одного з найдинамічніших саундтреків: музика в «Hades» підлаштовується під кожну сутичку. Саундтрек написав Даррен Корб, і він же налаштував у FMOD різні рівні інтенсивності треків. Під час дослідження підземного світу грає відносно спокійний ембієнт з

домішкою гітарних рифів, але при появі ворогів FMOD миттєво перемикається на «бойову» версію композиції: додаються важкі ударні, швидша мелодія. Коли бій закінчується, музика дочекається кінця такту і виконає фінальний акорд, синхронізований з останнім ударом, після чого знову повернеться до тихого фону. Така точність у музичних переходах стала можливою завдяки використанню маркерів і логіки в FMOD Studio, а інтеграція з рушієм (в даному випадку власним рушієм Supergiant на основі MonoGame) забезпечила зчитування ігрових подій (початок/кінець бою) без затримок. «Hades» часто наводять як приклад ідеальної реалізації адаптивної музики, що значно підсилює ігровий досвід.

Аудіо в сучасних іграх вийшло на такий рівень складності та важливості, що використання спеціалізованих аудіо-рушіїв (middleware) стало де-факто стандартом індустрії. Wwise і FMOD стали незамінними інструментами для звукорежисерів: вони надають можливості, які важко або неможливо реалізувати засобами лише рушія. На прикладах ігор ми бачимо, що правильно підібрані рушій та аудіо-middleware здатні створити по-справжньому інтерактивну музику, яка збагачує ігровий досвід. Інтеграція аудіомодулів дозволяє композиторам і звукорежисерам стати повноцінними учасниками процесу розробки, втілюючи свої творчі ідеї безпосередньо у фінальному продукті. У підсумку, технології Unity, Unreal Engine у поєднанні з Wwise, FMOD та подібними рішеннями відкривають перед ігровою індустрією широкі горизонти для подальшого розвитку адаптивного аудіо, роблячи ігри більш іммерсивними та емоційно насиченими для гравців.

2.2. Методи та інструменти технічної імплементації адаптивного музичного контенту.

У цьому підрозділі розглядаються методи та інструменти технічної імплементації адаптивної музики у відеоіграх. До ключових інструментів належать

спеціалізовані аудіо-middleware Wwise та FMOD, які виконують роль проміжної ланки між ігровим рушієм та музичним контентом і забезпечують керування інтерактивними музичними структурами в реальному часі. Основними методами є побудова параметричних моделей (RTPC у Wwise, Game Parameters у FMOD) для безперервного зв'язку ігрових змінних із параметрами звучання, використання станових моделей (States, Switches, Snapshots) для дискретного перемикання глобальних музичних режимів, а також організація музики у вигляді сегментів і шарів, що підтримують горизонтальне перемикання треків і вертикальне нашарування. Додатково застосовуються механізми тригерних подій та музичної синхронізації до тактової сітки, які дають змогу узгодити зміни саундтрека з геймплейними подіями, та засоби оптимізації аудіосистеми (стрімінг, ліміти голосів, компресія), що забезпечують стабільну роботу адаптивної музики на різних платформах. У подальшому тексті детально розглядається, як ці методи та інструменти взаємодіють між собою на етапах налаштування аудіо в middleware, інтеграції з кодом гри, тестування, оптимізації та реалізації конкретних сценаріїв динамічного музичного супроводу.

Після підготовки контенту музичні активи імпортуються в аудіосередовище (наприклад, Wwise чи FMOD), де звукорежисер визначає логіку їхнього варіювання. У middleware створюється ієрархія інтерактивної музики: музичні сегменти та шари організуються у контейнери, плейлисти або банки подій. На цьому етапі задаються правила переходів і параметри адаптації. Зокрема, використовуються параметричні моделі та стани аудіосистеми для керування музикою в реальному часі. Параметри (у Wwise зветься RTPC - Real-Time Parameter Controls, а у FMOD просто Game Parameters) дозволяють безперервно змінювати певні властивості звуку на основі змінних гри: від гучності окремих шарів до фільтрації або темпу [21]. Станова логіка (States у Wwise, або аналогічна механіка Snapshot'ів у FMOD) дає змогу перемикати глобальні режими звучання для різних контекстів гри (наприклад, «Спокій» vs «Бій»). Фахівець з аудіо налаштовує в middleware необхідні події (events), як-то подія старту фонові музики, події переходу на інший трек або вмикання

додаткового шару, та визначає, які параметри або стани вони змінюють. Наприклад, задаються музичні State-групи на кшталт Game State із можливими станами «Exploration» (дослідження) і «Combat» (бій), що впливають на вибір музичної теми та мікс шарів.

Наступний етап - з'єднання аудіосистеми з кодом гри, щоб змусити музику реагувати на ігрові події. Для Unity та Unreal Engine доступні офіційні пакети інтеграції Wwise та FMOD, які надають C#-скрипти, Blueprint-ноди та C++ API для керування звуком. Принцип роботи типовий: у кодї гри розробник запускає музичні події та оновлює параметри або стани залежно від геймплею. Код для Wwise (Wwise Audio Manager) постить музичну подію і перемикає стан музики або RTPC-параметр на основі подій гри. Програмісти можуть напряму змінювати States та RTPC із гри. Зв'язавши ці виклики з логікою ігрового процесу, досягається інтерактивність: музика буде автоматично змінюватися відповідно до поточної ситуації.

Завершальний крок - перевірка роботи адаптивної музики в грі та оптимізація продуктивності аудіосистеми. Необхідно ретельно відтестувати всі переходи та перемикання, щоб вони були музично плавними і доречними. Розробники перевіряють кросфейди між треками, синхронізацію зі змінами станів гри, відсутність затримок чи обривів звуку [**Ошибка! Источник ссылки не найден.**]. За потреби, параметри переходів підлаштовуються. Наприклад, збільшується час фейд-ауту або додаються stinger-звуки на зміну сцени, щоб згладити перехід. Одночасно слід стежити за тим, щоб складні адаптивні механізми не надто навантажували систему: велика кількість одночасно програних шарів або часте перемикання ефектів можуть впливати на продуктивність. Тому впроваджуються оптимізаційні заходи: стиснення аудіофайлів, стрімінг довгих треків з диску, ліміт голосів, тощо.

Наприклад, великі музичні файли доцільно стрімити з диска замість завантаження цілком у пам'ять. Це запобігає переповненню RAM [21]. Короткі звуки (саунд ефекти) зазвичай завантажуються повністю в пам'ять для швидкої реакції, а довгі фонові треки і музика - стрімінгом. Для уникнення

перевантаження звуку застосовується обмеження кількості одночасних голосів (звукових джерел): наприклад, дозволяється не більше певної кількості одночасних аудіо-потоків, менш важливі звуки автоматично пріоритизуються і заглушуються при перевищенні ліміту [там само]. Такі прийоми як управління пріоритетами, відстанню затухання звуку та пулінг аудіо-джерел дозволяють зберегти продуктивність без втрати якості звучання. Крім того, слід враховувати особливості платформи: для мобільних пристроїв з обмеженими ресурсами застосовується сильніше стиснення (OGG/AAC замість WAV), зниження частоти дискретизації, монофонічні доріжки для нетривимірних звуків тощо [там само]. Таким чином, успішна технічна імплементація адаптивної музики вимагає збалансувати багатий динамічний саундтрек із вимогами продуктивності системи [**Ошибка! Источник ссылки не найден.**].

Параметричні моделі в аудіосистемах визначають, як ігрові змінні впливатимуть на звук у режимі реального часу. Через RTPC гра може передавати числові значення (відсоток здоров'я гравця, рівень напруженості, швидкість руху тощо) до аудіомодуля, а той - відповідно змінювати параметри міксу або звучання. Наприклад, якщо здоров'я персонажа критично знизилося, гра може встановити RTPC «Health» на низьке значення, що в аудіосистемі прив'язане до ефекту пітчу та фільтрації: музика сповільнюється і спотворюється, додаючи відчуття небезпеки. Інший приклад: при наближенні до важливої цілі RTPC «Tension» поступово зростає, і це може підвищувати гучність тривожних музичних шарів та темп пульсації саундтрека [21].

Параметри можуть керувати гучністю шарів, частотними фільтрами, рівнем реверберації, панорамуванням, а також вибором варіантів музичних фрагментів. У FMOD Studio аналогічно використовуються Game Parameters з діапазоном значень, які можна прив'язати до будь-якого автоматизованого параметра міксу. Важливо, що RTPC дозволяють робити плавні зміни: у Wwise функція Set RTPC Value підтримує інтерполяцію значення параметра протягом заданого часу, автоматично згладжуючи переходи [55]. Це дає змогу, наприклад, поступово нарощувати інтенсивність музики (від 0 до 100%) протягом кількох

секунд, замість миттєвого стрибка. Звукорежисери визначають криві впливу RTPC на ті чи інші параметри звуку: наприклад, зробити так, що при значенні Intensity = 0 музика складається лише з базового спокійного шару, а при Intensity = 1 - звучать усі шари одночасно на повну гучність. Умовно, при Intensity = 0.5 буде проміжний стан: частина шарів звучиме на половинній гучності. Завдяки таким параметричним моделям забезпечується континуальний контроль над музикою - саундтрек реагує навіть на поступові зміни станів гри, а не лише на дискретні події.

Таким чином, параметричні моделі дозволяють гнучко прив'язати будь-який числовий геймплейний показник до змін музики: від простих (гучність, темп) до складних (динаміка еквалізації, глибина саундскейпу).

На додаток до безперервних параметрів, аудіоміддлвери надають механізм станів (State) та перемикачів (Switches) для реалізації дискретних змін аудіо контенту. States у Wwise - це глобальні режими, які визначаються певною State Group. Класичний приклад - група станів MusicPhase або GameState, що може приймати значення, як-от «Explore» (мирний стан), «Combat» (бій), «Stealth» (скритність) тощо. Кожен стан впливає на аудіо: можна призначити різні музичні сегменти або мікси для кожного стану. Коли гра змінює стан, аудіосистема автоматично переключає музику. Так, для стану «Combat» може бути визначено інший музичний трек або доданий агресивний шар, тоді як для «Explore» - спокійна тема без ударних.

Перехід між станами може бути налаштований із затримкою до музичної тактової сітки: Wwise дозволяє вказати, що перехід має відбутися не миттєво, а, наприклад, з початку наступного такту або з дограванням поточного фрагменту. Для цього використовуються Transition Matrix та Exit Cue в інтерактивних музичних сегментах. Крім того, у Wwise можна задати перехідні сегменти: спеціально скомпоновані короткі мости, що вставляються між музичними темами при зміні станів [21]. Це особливо корисно при складних переходах, щоб поєднати дві різні теми через загальний ритмічний чи гармонічний місток.

Аналогом State в FMOD є Snapshot або Global Parameter, що також може перемикатися між попередньо налаштованими міксами. Наприклад, у FMOD можна створити Snapshot «Combat», який піднімає гучність бойового шару і змінює еквалізацію, і викликати його активацію з коду при входженні в бій. Після завершення бою застосовується Snapshot «Calm» для повернення до спокійного міксу. Хоча різниця між перемикачами і станами технічно є (Switches у Wwise більше використовуються для вибору варіантів контенту на рівні окремих об'єктів), концептуально всі ці інструменти виконують схожу задачу: дискретне перемикання аудіо-контенту чи його властивостей відповідно до ситуації в грі. Станова модель зручна для сценаріїв, де музика повинна різко змінити настрій або тему при настанні певної події (початок бою, перемикання локації, перехід до фаз битви з босом тощо). У комплексних адаптивних системах часто використовують комбінацію станів і параметрів: стан визначає, яка саме тема грає, а параметри регулюють її інтенсивність чи варіації. Такий підхід застосовано, наприклад, у «Doom» (2016) чи «God of War», де глобальні стани відповідають різним рівням бойової ситуації, а параметри відображають градацію загрози і динамічно підлаштовують музику [21].

Для успішної імплементації адаптивної музики критичною є тісна синхронізація аудіо з подіями гри. Музична система повинна отримувати сигнали про геймплейні події миттєво і реагувати так, щоб зміни в музиці відповідали драматичному моменту, але водночас були музично коректними. Основні способи синхронізації включають:

Тригерні події гри: ігровий код надсилає повідомлення аудіосистемі при настанні певної події. Події можуть бути найрізноманітніші: початок/кінець битви, вхід у нову область, рівень загрози, зміна часу доби, досягнення чекпоінту в сюжеті тощо. Завдання програміста - в потрібний момент викликати відповідний метод аудіосервісу. Часто робиться спеціальний Audio Manager або Music Manager скрипт, що підписується на ігрові події (через Unity Events, делегати, або Blueprint події в Unreal) і виконує необхідні аудіо-реакції.

Музичну синхронізацію по такту/фразі. Щоб переходи не виглядали рваними, «аудіоміддлвер» може чекати відповідної ритмічної точки для зміни музики. Wwise підтримує синхронізацію переходів на початок наступного такту, тактового удару чи кінця музичної фрази через механізми «Transition Rules». Таким чином, навіть якщо подія надійшла в середині такту, новий фрагмент почне грати із невеликою затримкою, але точно з початку наступного такту – це збереже музичну цілісність. В випадках, де затримка небажана, використовують «кросфейд» - поступове накладення нового треку на старий без прив'язки до такту. Наприклад, при різкій зміні ситуації (раптовий напад ворога) можна миттєво запустити бойовий трек і плавно заглушити попередній протягом 1-2 секунд, якщо чекати музичного такту немає часу.

Стігери і one-shot ефекти – додатково до фонові музики, що безперервно грає, часто використовуються короткі музичні фрази або акценти - *stingers* - прив'язані до конкретних моментів (виграш битви, поява боса, отримання бонусу). Вони запускаються разово і накладаються на основний музичний фон, підкреслюючи подію. Їх реалізація аналогічна - тригер події в грі викликає відтворення окремого музичного Event в аудіосистемі.

Синхронізація тут критична по часу: затримка навіть у чверть секунди може зменшити драматичний ефект. Тому *stinger*-и зазвичай відтворюються негайно (без вирівнювання по такту).

Таким чином, синхронізація забезпечується як на рівні коду (вчасна генерація подій), так і на рівні аудіорушія (правильне музичне вирівнювання та переходи). Необхідно ретельно тестувати, щоб музичні зміни збігалися з візуальними подіями: уявімо, якщо музика бою ввімкнеться задовго до появи ворогів або, навпаки, із запізненням - це порушить іммерсію. Тому в процесі розробки аудіокоманда тісно співпрацює з геймдизайнерами, скриптерами: іноді доводиться навіть коригувати ігрові події (наприклад, додати невелику затримку перед кульмінацією сцени), щоб дати музиці «дихнути» і вчасно перейти на потрібний трек.

Адаптивна музика розширює можливості саундтреку, але водночас висуває додаткові вимоги до продуктивності гри. Множинні одночасні аудіопотоки, динамічне мікшування та ефекти можуть створити навантаження на CPU, пам'ять та диск. Тому оптимізація аудіо є невід'ємною частиною технічної імплементації. Основні напрями оптимізації були частково згадані раніше, розглянемо їх більш системно:

- *Управління пам'яттю і стрімінг:* музичні треки, особливо у високій якості, займають значні обсяги пам'яті. Завантажувати усі можливі музичні шари одночасно в RAM - неефективно. Краще використовувати streaming для довгих доріжок: дані зчитуються з диска поступово під час відтворення. Wwise та FMOD дозволяють помічати окремі файли як стрімінгові. Наприклад, основні петлі саундтреку можуть стрімитися, а короткі фрагменти (stinger-и) - бути у пам'яті для миттєвого програвання. Важливо підібрати розмір буферу і не перевантажувати диск великою кількістю одночасних стрімів, щоб не було затримок або «заїкання» музики.

- *Ліміт голосів і пріоритети.* Сучасний ігровий рушій може відтворювати десятки звуків одночасно (голоси). Без контролю це призведе до перевищення потужності: апаратно існують межі, скільки звукових каналів може змішуватися одночасно. Тому в аудіосистемі налаштовують максимальну кількість одночасних голосів (скажімо, 32 або 64). При перевищенні цього ліміту менш важливі звуки автоматично припиняються або не запускаються [21]. В контексті музики це рідше актуально (музика зазвичай займає 1-2 потоки), але якщо використовуються багатошарові саундтреки, кожен шар - окремий потік. Наприклад, 4-шаровий трек = 4 голоси. Якщо паралельно йде багато звукових ефектів, може спрацювати ліміт. Тому музичним потокам часто виставляють високий пріоритет, щоб вони не зрізалися. Також можна динамічно зменшувати кількість активних шарів у залежності від ігрової ситуації: наприклад, поза боєм відключати незначні шари, щоб зменшити навантаження.

- *Компресія аудіоформатів.* Для зниження навантаження на дискову підсистему та мережу (в разі стрімінгу) аудіо використовують у стислому вигляді. Музику в іграх зазвичай зберігають у форматах з втратами, як OGG Vorbis або AAC, які дають високу якість при меншому розмірі файлу. Це особливо актуально для консолей та ПК, де є достатньо ресурсів декодувати стиснене аудіо в реальному часі. На мобільних пристроях можуть використовувати нижчу частоту дискретизації (наприклад, 22 кГц замість 44 кГц) та моно звук для музики, щоб ще більше скоротити навантаження. Формати без втрат (FLAC) або некомпресовані (WAV) застосовують лише для коротких критичних звуків, де якість надважлива і є ризик артефактів від стиснення (наприклад, звуки інтерфейсу, якщо вони дуже короткі).

- *Оптимізація ресурсів аудіо движка.* Аудіо-middleware надають інструменти профілювання: в режимі відладки можна відстежити, скільки часу займає мікшування, скільки пам'яті використовується під банки звуків, скільки голосів активні тощо. Це дозволяє виявити «вузькі місця». Оптимізація може полягати у розбитті аудіобанків на менші (щоб не тримати в пам'яті неактивні звуки), відключенні непотрібних в даній сцені звукових підсистем, зменшенні складності ефектів (наприклад, рідше оновлювати дорогі DSP-ефекти або вимикати їх для далеких звуків). Часто доводиться балансувати між якістю адаптивної музики та витратами ресурсів. Наприклад, можна скоротити кількість шарів музики з 5 до 3, якщо продуктивність страждає, або зменшити довжину перехідних сегментів, щоб рідше їх завантажувати.

Отже, оптимізація - це невід'ємна частина роботи з адаптивною музикою. Правильно побудована система повинна бути масштабованою: працювати на різних платформах, від потужних консолей до мобільних пристроїв, адаптуючи якість аудіо під можливості заліза. Це досягається спільними зусиллями аудіопрограмістів та саунд-дизайнерів, які разом шукають компроміс між багатством звуку та швидкодією.

Розглянемо кілька поширених сценаріїв адаптивної музики у відеограх та способи їх технічної реалізації. Кожен приклад ілюструє застосування описаних вище методів у конкретній ситуації геймплею: параметрів, станів, шарів та ін.

Перший сценарій: у грі є спокійна фонова музика під час дослідження світу, але коли починається бій, музичний супровід повинен стати інтенсивнішим, драматичнішим. Після завершення бою саундтрек має плавно повернутися до спокійного настрою.

Найпростіший підхід - горизонтальне перемикання треків. Визначаються два окремі музичні сегменти – «Exploration» і «Combat». Через аудіо-стан GameState або перемикач, про які згадувалось вище, гра переключає активний сегмент. Наприклад, у Wwise для State Group «GameState» налаштовуються два стани: Exploration (відтворюється спокійний трек) і Combat (відтворюється бойовий трек). У коді при появі ворогів викликається SetState («GameState», «Combat») – і аудіосистема автоматично перейде на бойову тему з урахуванням заданих правил переходу (можливо, з фейд-аутом/фейд-ін або на початку такту). Після закінчення сутички аналогічно викликається SetState («GameState», «Exploration») для повернення до спокійної теми.

Інший, більш тонкий метод - вертикальне нашарування. В цьому випадку музика не переключається на іншу мелодію, а нарощує інтенсивність шляхом додавання шарів. Початково у спокійному стані грає тільки базовий шар (напр., атмосферна підкладка). Коли починається бій, гра збільшує значення параметра Intensity (або подібного) з 0 до, скажімо, 0.7. У відповідь аудіорушій поступово додає інші шари: ритмічну перкусію, швидший бас, активнішу оркестровку тощо. Такий підхід застосовано, наприклад, у саундтреці «Doom» (2016): під час бою музика складається з кількох шарів, які додаються у міру росту напруження бою, створюючи ефект наростання без зміни основної гармонії [34]. Технічно це реалізується за допомогою RTPC: кожному шару призначено криву залежності гучності від Intensity.

Обидва підходи можна об'єднувати: горизонтальне перемикання задає різні основи треків, а вертикальні шари додають динаміку всередині кожного

стану. Наприклад, при звичайному бою гра встановлює стан «Combat» з інтенсивністю 0.5 (середній бойовий трек з деякими бойовими шарами), а при появі боса - стан «CombatBoss» з інтенсивністю 1.0 (грає більш драматичний варіант теми з максимумом шарів). В результаті музика дуже точно відповідає розвитку подій: від тихого дослідження, через поступове нагнітання в малих сутичках, до повної «музичної люті» в кульмінаційних боях.

Розглянемо ще приклад. Сценарій: гравець переходить з однієї локації або рівня на інший, причому кожна локація має власну музичну тему. Потрібно забезпечити плавний перехід музики, щоб не було раптової тиші чи різкого обриву мелодії під час завантаження нового рівня.

Існує декілька технік вирішення цієї задачі. Поширений підхід - перекриття треків (crossfading). При виході зі сцени А музичний трек А не обривається відразу, а поступово згасає (fade-out) протягом кількох секунд. Одночасно у фоні починає програватися трек наступної сцени В, спочатку тихо, і нарощує гучність (fade-in) по мірі затухання попереднього. Якщо завантаження нової сцени відбувається непомітно для гравця (наприклад, без екрану завантаження), цей метод дозволяє музично «перемістити» гравця у нову атмосферу. У наведеному раніше коді

Більш складний прийом - музичні «стінгери» при переході. Композитор може написати короткий перехідний фрагмент, який грає саме в момент зміни сцени, слугуючи музичним мостом. Аудіосистема може автоматично вставити цей фрагмент: «Wwise», наприклад, дозволяє налаштувати «Transition Segment» між двома музичними темами [55]. Коли ігровий код сигналізує про зміну рівня (подія «Level End»), аудіосистема спочатку відтворює перехідний сегмент (наприклад, фінальна хвиля акордів, що логічно завершує тему А і готує тональність до теми В), і тільки потім запускає тему нової сцени. Такий метод особливо корисний, якщо дві музичні теми дуже різні: місточок допомагає зробити перехід більш естетично приємним.

Ще одна деталь - перенос музичного об'єкта між сценами. Іноді легше не зупиняти музику взагалі, а зробити об'єкт, що її програє, постійним між змінами

сцен (наприклад, «Unity» дозволяє позначити об'єкт як «Don't Destroy On Load»). Тоді музика просто продовжить грати без перерви при завантаженні нового рівня. Цей прийом доречний, якщо теми між сценами споріднені або плавно еволюціонують. У протилежному випадку (кардинально різні теми) все одно доведеться або кросфейдити, або різко переключити трек. Різке переключення прийнятне, якщо перехід супроводжується, наприклад, заставкою чи короткою паузою. Гравець сприйме зміну музики як частину зміни оточення.

Отже, при зміні сцен головне - уникнути музичного «дірки» (тиші) чи раптового невідповідного треку. Комбінація «fade out/fade in» та перехідних фрагментів, а також використання механізмів движка (Persistent audio between levels), забезпечує безшовність музичного оформлення і збереження емоційного тону гри навіть у моменти завантаження нових локацій.

Розглянемо ще один приклад. Сценарій: відкрита світова гра має цикл дня і ночі. Вдень навколишній світ спокійний і дружній, а вночі стає більш небезпечним або меланхолійним. Саундтрек повинен відображати ці зміни атмосфери: бути світлішим вдень і більш напруженим або сумним уночі.

Така адаптація може бути здійснена як через стани, так і через параметри. Простіший варіант - ввести стан «TimeOfDay» з двома значеннями: «Day» та «Night». У Wwise можна створити «State Group» «TimeOfDay» і задати для кожного стану свій мікс або навіть окремі музичні теми. Наприклад, в «Red Dead Redemption 2» саундтрек динамічно змінюється залежно від розташування гравця та часу доби: нічні поїздки супроводжуються більш приглушеною, задумливою музикою, ніж денні [Ошибка! Источник ссылки не найден.]. Це можна реалізувати, маючи базову тему, яка вдень і вночі аранжована по-різному. Аудіосистема при переході дня в ніч (ігрова подія Sunset або просто відстеження ігрового часу) переключає стан на «Night» - і, наприклад, додається нижчий стрій струнних, уповільнюється темп або тихо входить додатковий сумний шар. Вранці стан повертається на «Day», і нічні елементи поступово зникають.

Тонший підхід - використати безперервний параметр часу. Уявімо «RTPC TimeOfDay = 0» відповідає опівдню, «=1» - півночі (або навпаки). Гра постійно

оновлює цей параметр відповідно до внутрішнього годинника. Аудіорушій може прив'язати, скажімо, яскравість звучання або набір інструментів до цього параметра. До заходу сонця значення повільно зростає, і на певному порозі (0.7 наприклад) починають проявлятися нічні мотиви: приглушуються високочастотні інструменти, наростає легкий хор або падіння тону. Перевага цього способу - плавність змін: гравець практично не помічає, як музика з денного тема перетікає в нічну. Це як своєрідний кросфейд по часу: замість різкого перемикавання в певну годину, відбувається поступова зміна забарвлення протягом заходу сонця.

У технічній реалізації важливо, щоб гра подавала аудіосистемі точну інформацію про час або фазу дня. Це може бути подія (Day->Night, Night->Day) для станів або періодичне оновлення «RTPC» (наприклад, щохвилини ігрового часу оновлювати «TimeOfDay»). «Middleware» в свою чергу повинне бути налаштоване: якщо використовуються стани - задати правила переходу (можливо, з невеличким фейдом), якщо «RTPC» - визначити криві впливу на параметри міксу. Наприклад, можна автоматизувати фільтр еквалайзера: при значенні «Time_Of_Day» близькому до 1 (глибока ніч) трохи зрізати високі частоти, додаючи «темряви» в звук.

Результатом є динамічна музична підкладка, яка підсвідомо підказує гравцю про зміну часу доби. Вдень мелодія підтримує відчуття активності чи безпеки, а з приходом ночі - сигналізує про можливу небезпеку або викликає інші емоції (самотності, спокою - залежно від гри). Такий прийом підвищує іммерсивність: музика стає частиною живого світу гри, реагуючи навіть на цикл дня і ночі, а не тільки на дії гравця.

Відтак сценарій: музика гри змінюється не лише від стану бою чи часу, а й від стилю гри або успішності гравця. Наприклад, якщо гравець грає агресивно, музика насичується додатковими ударними; якщо діє обережно і непомітно - звучать тільки тихі атмосферні шари. Або залежно від того, скількох ворогів наразі залучено в погоню за гравцем, музика набирає більше шарів.

Реалізація сценарію в цьому типі адаптації зазвичай використовує параметри, пов'язані з поведінкою гравця. Розробники визначають метрики стилю гри - наприклад, коефіцієнт агресії, рівень шумності дій, кількість ворогів у радіусі, успішність (виграє чи програє бій) тощо - і прив'язують їх до музичних параметрів. Розглянемо декілька прикладів:

- *«Стелс vs Агресія».* У стелс-іграх часто реалізують музику, що підказує гравцю ступінь ризику. Коли гравець тихо прокрадається повз ворогів, звучить майже безмелодійний фон (напружений, але тихий). Якщо він починає діяти сміливіше або вороги починають щось підозрювати, шар інтенсивності збільшується – додаються пульсуючі ритми. При повному викритті (гравця помітили) музика переключається на бойову тему. Технічно можна використати параметр «StealthMeter» від 0 до 100, що відображає рівень помітності: при 0 - лише базовий шар, при 50 – додається напружений пульс, при 100 - тригериться стан бою або максимальний шар.

- *Кількість ворогів / рівень загрози.* Параметрична модель може напряму враховувати кількість противників, з якими одночасно бореться гравець. Наприклад, параметр «EnemyCount» використовується для мікшування: 0 ворогів - спокій, 1-2 вороги - легкий бойовий ритм, 5+ ворогів - повний набір бойових шарів. Реальний приклад: коли в грі «DOOM» інтенсивність музики підіймається з кожним убитим демоном: додаткові гітарні та ударні партії вмикаються по мірі того, як зростає «хаос» на екрані [34]. У «FMOD Studio» це можна налаштувати як параметр, що згладжено підвищує гучність кількох доріжок при збільшенні значення. На форумах «FMOD» розглядалася схожа задача: зробити так, щоб параметр «EnemyCount» «підмішував» різні шари музики відповідно до кількості переслідувачів гравця [21].

- *Стан здоров'я та успіх гравця.* Цей випадок схожий на розглянутий раніше з параметром здоров'я. Музика може підтримувати гравця, коли здоров'я низьке, або навпаки ставати більш тривожною. Також цікава механіка - адаптація до успіху чи невдачі в бою: якщо гравець домінує (знищує ворогів без втрат),

музика може ставати більш мажорною, гучною, «тріумфальною»; якщо навпаки - справи кепські, лунають мінорні мотиви, дисонанси. Це можна реалізувати через параметр «BattleMomentum» чи окремі стани «Winning»/«Losing». Горизонтальне перескладання теж підходить: наприклад, у грі може бути два варіанти бойової теми: для коли гравець бере гору і коли він притиснутий - і аудіосистема перемикається між ними залежно від співвідношення сил. Таке динамічне підлаштування під дії гравця робить досвід більш персоналізованим: музика ніби реагує на майстерність гравця.

З технічного боку реалізація змінних шарів вимагає від гри постійно відправляти оновлення відповідних параметрів. Часто це робиться в ігровому циклі: кожен кадр або з певною періодичністю гра рахує потрібні метрики і оновлює «RTPC». Аудірушій вже сам безперервно мікшує аудіо відповідно до поточного значення. Важливо налаштувати криві реакції: наприклад, щоб при зміні «EnemyCount» з 1 до 2 музика не перемикалася надто різко, вводять певний гістерезис або інерцію («Wwise» дозволяє задати інтерполяцію «RTPC» не миттєво, а за час). Це згладить аудіореакцію коли, скажімо, один ворог швидко вмирає і одразу з'являється інший.

Отже, адаптація шарів до поведінки гравця є по суті розширенням і комбінацією описаних підходів. Використовуючи параметри, що відображають різні аспекти дій гравця, можна досягти дуже тонкої емоційної підтримки геймплею музикою. Гравець несвідомо відчуває, що гра «відгукується» на його стиль (агресивний штурм або обережне планування) через музичні зміни. Це підсилює залученість і робить кожну ігрову сесію унікальною за звуковим супроводом.

Методи технічної імплементації адаптивного музичного контенту охоплюють широкий спектр інструментів: від підготовки самих музичних композицій, спроектованих під динамічні зміни, до використання спеціалізованих «middleware» (Wwise, FMOD) для реалізації цих змін у грі. Ключовими компонентами є параметри реального часу (RTPC), які забезпечують плавне реагування музики на ігрові змінні, та система станів (States), що дозволяє

швидко перемикає глобальні музичні настрої. В поєднанні з продуманою синхронізацією до ігрових подій і оптимізацією продуктивності, ці технології дають змогу створювати по-справжньому інтерактивні саундтреки. Музика більше не є статичним фоном - вона перетворюється на активний елемент ігрового процесу, реагуючи на дії гравця і підсилюючи емоційне сприйняття кожного моменту гри.

2.3. Композиційні стратегії створення адаптивних музичних структур для ігор.

У дослідженні адаптивної музики під стратегіями створення розуміються композиційно-технічні підходи до організації саундтрека, які визначають, як саме музика буде змінюватися під впливом геймплею. До основних стратегій належать: вертикальне нашарування (*vertical layering*), горизонтальне перемикання та монтаж сегментів (*horizontal resequencing*), гібридні моделі, що поєднують вертикальні й горизонтальні підходи в єдиній системі, а також генеративні та процедурні стратегії, у межах яких музика частково або повністю синтезується алгоритмічно в реальному часі на основі заданих правил. Подальші історичні приклади демонструють еволюцію та практичну реалізацію цих стратегій у конкретних іграх.

Одним із перших проривів у сфері інтерактивної музики стала система iMUSE, розроблена композиторами Майклом Лендом і Пітером Макконнеллом у «LucasArts». Вона вперше застосована в грі «Monkey Island 2: LeChuck's Revenge» (1991) і забезпечила безшовну зміну музики залежно від дій гравця [56]. У «Monkey Island 2» музика плавно підлаштовується під переміщення героя: коли гравець переходить з однієї локації до іншої, оркестровий фон змінюється, але перехід майже непомітний. Секрет полягав у тому, що iMUSE могла додавати або прибирати інструменти на льоту, а також знала, на якому такті йде мелодія, щоб у потрібний момент вставити спеціальний музичний

брідж (короткий перехідний фрагмент). Ці бріджі згладжували стики між різними фрагментами саундтреку, зливаючи їх в одне цілісне виконання. Як згадував Пітер Макконнелл, ідея була зробити музичну підсистему «розумнішою», перетворивши її на подобу диригента, що реагує на дії гравця: «музика мала працювати як оркестр в оркестровій ямі, а не як грамофон, де просто ставиш голку - мелодія змінюється, щойно щось сталося» [цит. за 56].

Вплив iMUSE на індустрію був значним. Після «Monkey Island 2» багато проєктів почали активно експериментувати з динамічним озвученням. Наприклад, у «Ultima Underworld» (1992) Дж. Сангера музика реагувала на ігрові ситуації, а у Quest for Glory V (1998) композитор Чанс Томас використав адаптивний оркестровий саундтрек [55]. Особливо показовим став підхід студії «Rare» у грі «Banjo-Kazooie» (1998). Композитор Грант Керкхоуп, надихнувшись ідеями «LucasArts», реалізував контекстне аранжування: кожна зона рівня мала власну варіацію головної теми. В саундтреку було передбачено кілька шарів-інструментів (через MIDI-треки). Наприклад, у звичайній області звучить повна версія мелодії, а під водою автоматично відключаються високі інструменти і додається ефект приглушення, створюючи підводну атмосферу [56]. Перехід здійснювався через невидимі тригери: коли персонаж перетинав певну точку на мапі, двигун гри перемикав потрібні музичні доріжки. Цей метод вертикального мікшування забезпечив плавне й непомітне для гравця змінення аранжування. Музика ніби «дихала» разом з ігровим світом.

З розвитком технологій і зростанням обсягів пам'яті композитори дістали змогу використовувати згаданий вище прийом вертикального нашарування у повному масштабі. Ранній приклад - саундтрек Мартіна О'Доннелла до серії «Halo». Ще для «Halo: Combat Evolved» (2001) О'Доннелл створив власний аудіорушій, оскільки на той час не існувало готових рішень для інтерактивної музики. Він заклав основи системи, яка підтримувала рандомізацію фрагментів, накладення шарів і реакцію на дії гравця [40]. Саундтрек «Halo» складався з декількох рівнів інтенсивності: спокійний ембієнт у моменти дослідження, напружені ритми під час бою тощо. Коли починалася перестрілка, гра поступово

«нарощувала» аранжування - додавала ударні, хорові партії чи струнні на фон основної теми. Музика адаптувалася в реальному часі, залишаючись епічною і цілісною - це було новаторством на початку 2000-х.

Ще більшого розквіту вертикальне мікшування досягло у сучасних відкритих світах. Наприклад, саундтрек «Red Dead Redemption» (2010) та «RDR 2» (2018) композитора Вуді Джексона побудовані майже повністю на принципі багат шаровості. Уся музика там була записана у єдиному темпі й тональності (для першої частини - 130 ударів на хвилину, ля мінор) [56], а потім розбита на окремі інструментальні теми: гітари, скрипки, перкусія, тощо. Ігровий рушій міг у режимі реального часу комбінувати ці сегменти різними способами. Саме тому єдиний саундтрек покривав і драматичні перестрілки, і самотні мандрівки пустелею, не звучачи при цьому невлад [там само]. Джексон описував свій метод як написання «великих музичних полотен» тривалістю до 20 хвилин, з яких гра вже сама «нарізає» потрібні фрагменти залежно від ситуації: «це і називається динамічна музика - партитура змінюється відповідно до дій гравця» [**Ошибка! Источник ссылки не найден.**]. В «Red Dead Redemption 2», до речі, музика реагує навіть на моральність героя: саундтрек буде більш меланхолійним чи тріумфальним залежно від того, як гравець поводить у грі [53].

Схожі вертикальні стратегії використовуються і в інших блокбастерах. Саундтрек «The Witcher 3: Wild Hunt» (2015) змінюється під час бою: до мелодії локації додаються додаткові шари (ударні, вокал), наростає темп, а після перемоги музика повертається до спокійного фону. Битви у грі мають кілька рівнів інтенсивності, між якими саундтрек перемикається залежно від напруженості сутички [53]. Такий ефект досягається завдяки багат шаровим композиціям: основна тема кожного регіону гри має варіації: базову (спокійну) і бойову (збагачену драйвом). Перехід здійснюється м'яко, через підміксування або витягування окремих треків. Подібний прийом ще раніше використовував Джеспер Кід у «Hitman: Blood Money» (2006), де при переході в режим тривоги до саундтреку додавались агресивніші оркестрові елементи, а в стелсі музика ставала мінімалістичною.

Розглянемо приклад структури адаптивного саундтреку на основі саундтреку «DOOM Eternal». Залежно від перебігу бою музика переходить між станами: від спокійного ембієнту (Ambient) до кульмінаційного «хору» (Chorus) при піковій напрузі. Плавні переходи забезпечуються через спеціальні вступні фрази (Intro) та наростаючі секції (Build-up), що з'єднують фрагменти без швів [34].

Одним із найдосконаліших прикладів вертикального нашарування є музика Міка Гордона до цієї гри. Гордон застосував модульний підхід: написав десятки музичних петльових фрагментів і розділив їх на тематичні теми: окремі доріжки ударних, басів, гітар, синтезаторів, хору тощо [34]. У грі ці доріжки змішуються на льоту. Спеціальні скрипти слідкують за обстановкою: скільки ворогів навколо, наскільки агресивно діє гравець, чи отримує він ушкодження – і відповідно нарощують або зменшують густоту музики. Якщо герой нишпорить рівнем без бою, звучить майже безритмовий темний ембієнт із мінімумом перкусії (Ambient Layer). Коли з'являються демони, система поступово додає шари: спочатку легкий бойовий ритм (Light Combat - прості гітарні рифи, барабани), далі середній (Medium Combat - агресивніші рифи, швидша перкусія), і нарешті максимальний рівень (Heavy Combat - швидкі ударні, «важкі» гітари, шумові ефекти) для найзапекліших сутичок. Всі ці фрагменти були заздалегідь скомпоновані в одній тональності та близькому темпі (близько 120–140 BPM), тому гра може безшовно перемикатися між ними. Перехідні сегменти синхронізуються через middleware, що дозволяє уникнути різких стрибків гучності чи тональності. Результат - неймовірно «живий» саундтрек, який підлаштовується (адаптується) під кожен сплеск екшену. «DOOM» задав нову планку адаптивності: гравці відчувають, як музика «пульсує» в унісон з боєм, додаючи адреналіну [34].

Метод горизонтальних переходів і тематичних варіацій використовується для побудови музичних переходів між сценами чи станами. Наприклад, у «Journey» (2012, композитор Остін Вінторі) вся партитура була задумана як єдина протяжна тема, яка розвивається у ході мандрівки героя. Основний мотив

(віолончель) символізує гравця, інші інструменти - світ і події довкола нього. Коли гравець досягає певних ключових точок, гра перемикає музичні сегменти: так, у фінальному рівні (трек «Apotheosis») оркестровий розвиток тимчасово зупиняється і повторюється в циклі, доки гравець не дійде до вершини гори. Лише після цього вступає наступна частина твору (стриманий вступ скрипки). Це горизонтальна адаптація: музика очікує дії від гравця для переходу до наступного розділу композиції. Завдяки такому підходу саундтрек Journey кожного разу звучить трохи інакше: темп появи нових інструментів і кульмінацій залежить від темпу проходження гри. При цьому Вінторі зумів зберегти емоційну цілісність музики: за його словами, головне – «дати гравцеві максимум свободи, але не жертвувати музичною драматургією» [56]. «Journey» вдало поєднує горизонтальний і вертикальний методи, демонструючи, що адаптивна музика може бути такою ж виразною, як і лінійна.

Горизонтальний підхід часто застосовується для інтерактивних музичних стиків - коротких фрагментів, що сигналізують про подію. Композитори ретельно прописують такі переходи в партитурі, часто готуючи кілька варіантів закінчень. Приміром, у рольовій грі «Undertale» (2015) Тобі Фокс створив альтернативні версії бойових тем, що залежать від шляхів гравця: мелодія битви з босом може суттєво відрізнятись у пацифістському і геноцидному проходженні, відображаючи моральний вибір [53]. Це зроблено шляхом горизонтального розгалуження саундтреку - гра запускає різні музичні файли під час ключових моментів сюжету. Схожа концепція в «NieR: Automata» (2017): кожна локація має кілька рівнів аранжування (від тихого соло до повноцінного оркестру), а окремі треки мають вокальні версії, що вмикаються при повторному відвідуванні з новими сюжетними обставинами. Композитор Кейічі Окабе, по суті, написав для багатьох композицій кілька розвитків наперед, і гра вибирає потрібний залежно від контексту.

Генеративна та процедурна музика. Окремим напрямом є алгоритмічне генерування музики в реальному часі на основі правил (procedural music). Якщо вертикальні й горизонтальні методи оперують заздалегідь підготовленими

фрагментами, то генеративні системи компонують саундтрек «на льоту» з наборів заданих мотивів, патернів і умов. Цікавою реалізацією є саундтрек до медитативної гри «No Man's Sky» (2016). Більшість музики в ній - не статичні треки, а результат роботи внутрішньої програми під назвою Pulse, створеної звукорежисером Полом Віром у співпраці з пост-рок гуртом «65daysofstatic» [56]. Вір задав системі контекстні правила: для кожного типу середовища (відкритий космос, поверхня планети, бойова ситуація тощо) визначено набір музичних елементів і параметрів їхнього варіювання. Наприклад, якщо гравець різко розвертає корабель у космосі в бік планети, гра може згенерувати плавне наростання арпеджіо або додати новий ритмічний шар. Але відбувається це не завжди, випадковість теж передбачена. Такий дизайн гарантує, що саундтрек ніколи дослівно не повторюється, він «вишикується» із пазлів залежно від дій гравця. Тим не менше, ключові моменти (наприклад, перший виліт у космос чи знаходження важливої локації) підтримуються заздалегідь написаними вставками: «No Man's Sky» комбінує процедурність із традиційною композицією, щоб важливі сцени мали емоційно чітку музичну тему. Пол Вір називає такий підхід «музичним садом можливостей»: композитор закладає насіння у вигляді тем і правил, а гра вирощує з них неповторні варіанти звукового ландшафту [56].

Ще один приклад тонкого процедурного підходу - «Jessica Curry - Everybody's Gone to the Rapture» (2015). В цій грі кожна локація (село, ферма, ліс тощо) має власний набір інструментів і алгоритм, що визначає їхнє увімкнення. Чим далі гравець просувається у розкритті сюжету області, тим щільнішою стає музична фактура: скрипт поступово підвищує «стан» області, дозволяючи «частіше прозвучати додатковим інструментам» [56]. На початку гра може ледь чутно відтворювати окремі ноти фортепіано або голосу, а ближче до розв'язки цієї глави - розгортається повний хор і струнні, підкреслюючи наростання напруги. При цьому переходи настільки плавні, що гравець майже не усвідомлює, як музика від легкого тла перейшла у багатоголосий ембієнт - емоція зростає підсвідомо. Композиція будується алгоритмом, але з

використанням підготовлених композитором фраз; це ніби імпровізація, керована параметрами геймплею.

Процедурні методи особливо доречні в іграх з неструктурованим, хаотичним процесом. Яскравий випадок - ритмічний екшен «Are Out» (2019), саундтрек якого генерується у стилі фрі-джаз. Партія ударних формується у реальному часі: кожен удар, постріл чи кидок супротивника тригерить звук барабана або тарілки. Гравець сам того не знаючи виступає «виконавцем» - ніби невидимий джазовий перкусіоніст супроводжує його дії. У результаті музика і гра зливаються: хаотичні удари створюють драйвовий саундтрек, що ні секунди не повторюється і ідеально збігається з екшеном на екрані.

Інша нетипова реалізація - гумористична «Untitled Goose Game» (2019), де адаптивна музика побудована на класичному творі. Композитор Ден Голдінг узяв фортепіанну прелюдію Клода Дебюссі, розбив його на шість фрагментів і реалізував систему, яка міксує ці фрагменти залежно від поведінки гравця-гуски. Коли гусак нишпорить спокійно - звучать тихі уривки, якщо здіймає гармидер - гра підхоплює жвавіші розділи прелюдії. Це приклад горизонтально-вертикального гібриду: готовий музичний твір трансформовано у інтерактивну форму, але при цьому збережено живу динаміку оригіналу. Саундтрек сприймається як жива джазова імпровізація піаніста, що реагує на витівки гуски.

Як видно з наведених прикладів, сучасні ігри рідко обмежуються якоюсь однією стратегією адаптивної музики. Найчастіше вертикальний і горизонтальний підходи поєднуються: гра може одночасно міксувати шари й переставляти фрагменти. Наприклад, у «Halo Infinite» (2021) саундтрек реагує і на зміну бойових фаз (горизонтально переключаючи музичні теми між сутичками та спокоєм), і на рівень загрози (вертикально додаючи інструменти при появі сильних ворогів). Завдання композитора - передбачити всі можливі стани геймплею і спланувати музичні елементи так, щоб вони логічно поєднувалися у будь-якій комбінації. Майкл Світ у книзі «Writing Interactive Music for Video Games» зазначає, що часто найкращий результат дають саме гібридні підходи, де горизонтальні і вертикальні методи підтримують один

одного [45]. Вертикальне нашарування забезпечує плавність і безперервність саундтреку, тоді як горизонтальне перемикання - більш виразні зміни, прив'язані до подій гри.

Композитори виробили низку практичних прийомів, які гарантують успішне впровадження адаптивної музики. По-перше, єдність темпу і тональності у взаємозамінних фрагментах (як ми бачили у прикладі «Red Dead Redemption», використання єдиної гармонічної основи дозволяє легко з'єднувати різні частини без дисонансу) .По-друге, написання музики шарово: навіть працюючи з оркестром, композитор мислить категоріями окремих груп (струнні, духові, ударні, хор тощо) і продумує, як кожен шар звучатиме ізольовано. Такий підхід допомагає і при тестуванні - ще до інтеграції в гру різні комбінації STEM-доріжок прослуховуються, щоб переконатися, що вони сумісні і при накладанні, і при кросс-фейді. По-третє, важлива музична маркеризація: на стиках фрагментів композитор вставляє спеціальні перехідні такти або ефекти (наприклад, затухаюче дзвоніння чи удар гонга), щоб закрити момент зміни (так робили ще в iMUSE, і це актуально досі) [56]. По-четверте, інтерактивна музика тісно координується зі звуковим дизайном гри. Як образно каже Мартін О'Доннелл, «звук робить світ реальним, а музика - емоційним» [40]. Щоб не перевантажувати слухача, потрібно знаходити баланс: іноді краще додати атмосферних шумів замість музики (як у хоррорах), а іноді - навпаки, приглушити звуки на користь саундтреку в драматичний момент.

Підсумовуючи, адаптивні композиційні стратегії відкрили для ігрової музики новий вимір. Сьогодні саундтрек - це не зафіксована фонограма, а гнучка, жива частина ігрового світу. Композитори використовують весь арсенал прийомів: модульні структури, паралельні шари, алгоритми й рандомізацію - щоб музика підлаштовувалася під кожну дію гравця. Успішні кейси на кшталт «DOOM», «Red Dead Redemption 2», Journey чи Halo демонструють, що інтерактивна музика здатна не лише посилювати емоції, а й вибудовувати цілісний наратив разом з грою. Адаптивні саундтреки стали стандартом у AAA-проектах і все частіше з'являються в інді-іграх, збагачуючи ігровий досвід

новими гранями. З огляду на темпи розвитку аудіотехнологій, можна прогнозувати подальше розширення можливостей: ще більш тонке підлаштування під поведінку гравця, генеративні музичні системи зі штучним інтелектом, персоналізація саундтреку під вподобання користувача тощо. Але вже зараз очевидно, що адаптивна музика стала невід'ємною частиною ігрового мистецтва, піднімаючи занурення у віртуальні світи на недосяжний раніше рівень.

РОЗДІЛ 3. ПРАКТИЧНА РЕАЛІЗАЦІЯ ТА АНАЛІЗ АДАПТИВНОГО МУЗИЧНОГО ОФОРМЛЕННЯ ДЕМО-ПРОЄКТУ

3.1. Концептуальне обґрунтування музичного оформлення обраного ігрового демо-проєкту.

Обраним демо-проєктом для практичної реалізації адаптивної музики стала навчальна гра «Gem Hunter Match» - казуальна головоломка жанру match-3 з атмосферою пригод і легкої магії. Візуальний стиль гри - яскравий мультиплікаційний 2D, а ігровий процес передбачає збирання коштовностей та вирішення пазлів. Відповідно до художнього задуму, музичне оформлення має органічно доповнювати цей ігровий світ, підсилюючи відчуття дослідження чарівних підводних локації і скарбниць. Музика в відеогрі - це не просто тло; вона формує атмосферу, впливаючи на емоції гравця та занурюючи його у світ гри [52].

Саундтрек гри буде структуровано в кілька станів, що відповідають різним етапам ігрового процесу: фоновий режим, активна гра, перемога або поразка та в подальшому плануванні - режим з таймером (напруження). У спокійні моменти, коли гравець не обмежений часом і просто досліджує або робить ходи без поспіху, гратиме основна фонові тема (мінімальна інструментовка, м'який ритм). Ця музика створюватиме відчуття магічної атмосфери і безперервності світу, але не відволікатиме від обдумування ходів. Коли гравець не робитиме

ніяких дій - програватиметься основна тема в обрізаному варіанті. Варто зазначити, що в нинішньому варіанті гри рівнів з таймером поки немає, але в майбутньому у режимі таймера музика переключатиметься на спеціальну напружену тему. Ця тема прямо вказуватиме на зростання напруги та важливість моменту, підвищуючи у гравця відчуття терміновості. В момент виграшу чи програшу програватиметься відповідний стінгер. Подібний підхід відповідатиме усталеній практиці динамічних саундтреків, де саундтрек адаптується в реальному часі до ситуації в грі - наприклад, переходить від «спокійної» музики до «напруженої» при появі загрози чи обмеження в часі [52]. Ще у класичних іграх було закладено таку концепцію музичних станів (спокійна тема vs. «бойова» чи аварійна при зміні обставин), а сучасні технології дозволяють реалізувати їх плавно й непомітно.

Також варто окреслити музичну форму, оскільки вона визначає, як композиція функціонує всередині ігрового процесу. Основна фонова тема матиме одночастинну зациклену форму, вибудовану як безперервна музична петля. Це дозволить їй звучати довго, не перевантажуючи увагу та створюючи стабільне відчуття магічної атмосфери. Тема для режиму з таймером буде не окремою композицією, а варіацією основної теми, зосередженою на підвищенні напруги: вона зберігатиме ключові мотиви базового лупу, але матиме виразніші ритмічні акценти, густішу фактуру та відчуття наростаючої терміновості. Завдяки такому варіаційному підходу музика залишатиметься впізнаваною, проте емоційно підсиленою саме для критичних моментів геймплею. Стінгери ж виконуватимуть роль коротких джінглів, які миттєво сигналізуватимуть про перемогу чи поразку й слугуватимуть чіткими емоційними маркерами. У сукупності така формальна організація дозволить саундтреку плавно та природно вплітатися у структуру гри: основна тема забезпечувати спокій і безперервність, варіація для таймера – драматичну напругу, а стінгери - швидку емоційну реакцію, що суттєво підсилить імєрсивність.

Основна музична тема повинна передавати відчуття дослідження і легкої чарівності, що відповідатиме казковому антуражу гри. Обрана стилістика -

світла, мелодійна оркестрова музика з казковими мотивами - покликана викликати у гравця відчуття пригоди та відкриття. Такий підхід узгоджується з ігровим нарративом і візуальною стилістикою, роблячи саундтрек невід'ємною частиною світу гри. Вдалий саундтрек здатен зробити ігровий процес спокійним й захопливим, формуючи емоційний зв'язок гравця з ігровим середовищем. Обраний музичний стиль також підтримуватиме геймдизайнерські елементи: наприклад, мотив скарбошукача (пригодницький настрій, чарівні дзвіночки) підкреслюватиме тему збирання коштовностей та магічних предметів, а казкове оркестрове звучання доповнить яскраву мультиплікаційну графіку, підсилюючи аудіовізуальну цілісність і занурення гравця у гру. Відомо, що невідповідний або розірваний музичний супровід може порушити це занурення: як зазначає К. Коллінз, «розрізнена партитура зазвичай призводить до розрізненого ігрового досвіду» [13]. Тому стиль і темп музики були ретельно узгоджені з тематикою локацій та подій гри, аби забезпечити емоційну конгруентність звуку й дії.

Щоб уникнути втомлюваності від музики, саундтрек має бути ненав'язливим: помірний темп, збалансована динаміка, відсутність різких дисонансів чи надто нав'язливих мотивів. Дослідження підтверджують, що надто монотонна або нав'язлива музика швидко стомлює гравців при тривалих сесіях [52]. Тому мають бути використані м'які переходи та варіативність аранжування для підтримання інтересу: зокрема, передбачено перемикання або приглушення окремих шарів інструментів у різних циклах. Така мікроваріативність (на кшталт вмикання/вимикання певних інструментальних партій, зміни насиченості фонового акомпанементу тощо) є дієвим засобом утримати увагу гравця і запобігти відчуттю нав'язливого «заїждженого» лупу. Іншими словами, навіть базова фонова музична тема міститиме внутрішню динаміку та розвиток, лишаючись комфортною для багаторазового прослуховування.

Стінгери виграшу та програшу виконуватимуть функцію коротких аудіосигналів, що забезпечують оперативний зворотний зв'язок і підсилюють емоційну частину ігрового досвіду [**Ошибка! Источник ссылки не найден.**]. Їхня головна мета полягатиме у швидкому та однозначному позначенні

результату дії гравця, водночас збереженні загальної стилістичної цілісності звукового середовища. На відміну від фонового музичного оформлення, такі стінгери не формуватимуть тривалий наратив, а діють як виразні маркери події, які миттєво кодують успіх або невдачу.

Стінгер виграшу ґрунтуватиметься на мажорній гармонії та світлих тембрових рішеннях, оскільки його завданням є створення відчуття завершеності, винагороди та легкого емоційного підйому. У мультяшній естетиці для цього часто використовуються дзвінки тембри на кшталт ксилофона, марімби або синтезованих перкусійних інструментів, що асоціюються з легкістю та дитячою казковістю. Інтонаційна структура матиме висхідний характер, що підсилюватиме відчуття прогресії та позитивної динаміки. Тривалість стінгера буде мінімальною, щоб не заважати подальшим системним звукам і не порушувати темпоритм інтерфейсних анімацій.

Стінгер програшу, навпаки, базуватиметься на мінорних інтонаціях, які зформують легке відчуття падіння, зниження енергії або «комічної невдачі». У мультяшних казуальних іграх характерною є м'яка, «безпечна» форма звукової поразки, яка не створює надмірно негативного впливу на користувача. Тембр може бути більш приглушеним або «пухким», часто з використанням коротких низхідних глісад чи невеликих портаменто, що сформують ефект спаду. Таке рішення допоможе підкреслити результат без емоційного перевантаження, що особливо важливо для казуального сегмента, де ігровий досвід має залишатися легким і ненапруженим.

Обидва типи стінгерів виконуватимуть важливу когнітивну функцію - вони полегшать сприйняття гравцем причинно-наслідкових зв'язків та сприятимуть формуванню звичок у ході ігрового процесу. Завдяки стабільності звукових патернів гравець швидше ідентифікує успішні або помилкові дії та адаптується до механік. У межах мультяшної казуальної естетики стінгери також підтримуватимуть загальну казкову та добродушну атмосферу, забезпечуючи баланс між інформативністю та емоційною нейтральністю [52].

Всі зазначені стани та варіанти музики буде втілено у інтерактивній формі завдяки використанню «middleware FMOD», інтегрованого в «Unity». Технічна реалізація дозволить адаптивно керувати саундтреком: гра постійно відслідковуватиме параметри геймплею і запускатиме відповідні музичні події. Буде застосовано принципи реального часу адаптації звуку, коли музика змінюватиметься динамічно залежно від дій гравця та стану ігрового середовища. Як наслідок, саундтрек виступить своєрідним музичним «фідбеком» на ігрові події, підсилюючи відчуття контролю та залученості гравця.

3.2. Процес створення та продакшну адаптивного музичного контенту: композиційні та технічні рішення.

Обидві оригінальні композиції та стінгери (додаток А), створені для гри «Gem Hunter», мають конкретне призначення та характер у контексті геймплею. Перша композиція слугує фоновим музичним супроводом на звичайному рівні: вона спокійна, мелодійна, циклічна (закільцьована для безперервного повторення). Другий трек - інтенсивна тема для запланованого в майбутньому режиму з таймером, що звучить напружено, ритмічно та більш насичено, підкреслюючи відчуття терміновості у ситуації обмеженого часу. Треки написані у секвенсері Steinberg Cubase із використанням сучасних віртуальних бібліотек інструментів та семплів: «Arcade» від Output, «Cinestrings», «Spitfire Chamber Strings», «Cinewinds», «Berlin Strings», «Action Strikes» від Native Instruments, «Omnisphere» від «Spectrasonics». Застосування цих бібліотек забезпечило високоякісне оркестрове звучання та широкий спектр тембрів [21]. Вже на етапі написання музики було передбачено інтерактивне використання: композиції спроектовані для горизонтальної секвенсії з можливістю вертикального нашарування (розділені на окремі інструментальні стемі)(додаток Б) і витримані в єдиній тональності для зручного поєднання та перемикання через FMOD.

Такий підхід відповідає сучасним принципам адаптивного саундтреку, коли всі музичні елементи узгоджені між собою і можуть комбінуватися без гармонійних конфліктів.

Першим етапом роботи над музичним супроводом «Gem Hunter» став аналіз ігрового процесу. Було відкрито прототип гри, проведено кілька тестових проходжень з метою оцінити динаміку геймплею, характер рухів та загальний ритмічний темп. Для визначення оптимального темпоритму було відібрано референсний музичний трек, після чого здійснено серію експериментів із його прискоренням та уповільненням. Це дозволило встановити темп, який найбільш органічно відповідав часовим і моторним відчуттям від гри та забезпечував гармонійне поєднання музики з геймплеєм.

Подальший творчо-технічний процес будувався планомірно, за принципом поетапної структуризації музичного матеріалу. В основу першої теми був покладений базовий ритмічний «каркас», сформований партією контрабаса, що задала фундаментальний ґрув та гармонічну опору композиції. Наступним кроком було створення гармонічного супроводу засобами піцикато струнних і партії металофону, які визначили фактурну щільність та ритмічну артикуляцію твору. На сформовану гармонічну й ритмічну структуру була накладена основна мелодична лінія у тембрі флейт, підібрана таким чином, щоб підтримувати світлий, магічний характер ігрового середовища. Паралельно здійснювався добір текстурних елементів: за допомогою бібліотеки «Arcade» by Output було інтегровано м'які ембієнтні пади та рухливі атмосферні шари, що забезпечили просторову глибину. На ранньому етапі було проведено попередній баланс інструментів, що дозволило оцінити взаємодію тембрів і визначити корекції для уникнення надмірного перекривання фактури.

Створення другої композиції (теми для майбутнього режиму з таймером) здійснювалося на основі першої, але з істотною трансформацією. Збережено темброву спорідненість та композиційний задум, однак темп було підвищено з метою посилення відчуття напруженості. Ритмічну основу було збагачено перкусійними елементами з колекції Action Strikes, які забезпечили

імпульсивність та підкреслили обмеження часу як геймплейну механіку. Додаткові мікродеталі (короткі підготовчі ноти, фактурні swell-ефекти та допоміжні флейтові підголоски) сприяли оживленню текстури та підсилювали експресивність композиції.

Створення стінгерів здійснювалося як окремий етап композиційної роботи. Стінгер перемоги було побудовано на матеріалі основної теми: він являє собою короткий мажорний джінгл, що поєднує фінальний акорд із плавним струнним жестом, формуючи відчуття завершеності та позитивного результату.

Музичні теми написані в тональності до мажор (C Major), що визначає її світлий, мажорний настрій. У гармонії переважають діатонічні акорди натурального мажору, зокрема, стійкі функції тоніки (C), субдомінанти (F) та домінанти (G) формують основу прогресії, підкреслюючи консонансність і піднесений характер звучання. В цілому гармонічна структура є простою і циклічною, що характерно для тем, призначених багаторазово повторюватися у інтерактивному середовищі (ігровому саундтреку).

Особливої уваги заслуговує поява однієї нетипової для до мажору ноти - А \flat (ля-бемоль). Це понижена шоста ступінь, яка не входить до складу натурального мажорного ладу. Її використання можна пояснити явищем модального змішування (modal interchange) - запозиченням із спорідненої тональності (до мінору) або альтернативного ладу. Застосування такої хроматичної ноти привносить легкий магічний відтінок та різноманітність у гармонію, не руйнуючи загальну тональність [**Ошибка! Источник ссылки не найден.**]. Зокрема, А \flat може виступати як частина запозиченого акорду: наприклад, субдомінанти у мінорі (Fm - фа мінор) або акорду \flat VI ступеня (А \flat major) у мажорі. Подібний borrowed chord (запозичений акорд) - А \flat мажор - є досить поширеним прийомом у C мажорі; він додає несподіваний колорит, оскільки знаходиться поза межами основної гами, але водночас достатньо близький до тональності, щоб сприйматися природно. Таким чином, поява ля-

бемоля виконує функцію кольорового гармонійного відхилення, яке збагачує мажорну тему, зберігаючи її світлий характер.

Для забезпечення збалансованого звучання оркестрового треку було застосовано ряд технік обробки при мікшуванні, а також ретельний мастеринг фінального матеріалу.

Для легато-інструментів, що ведуть мелодію (наприклад, флейти), використано м'яку компресію за допомогою оптичного компресора «Waves CLA-2A» (додаток В). Він делікатно вирівнює динаміку мелодичних ліній, приборкуючи піки гучності та вирівнюючи рівень тихіших звуків. Такий підхід дозволяє зберегти плавність легато і рівномірність звучання мелодії без відчутного «стискання» [45]. Додатково застосовано еквалайзери: «FabFilter Pro-Q 4» (додаток Г) для точної корекції частотного балансу та «Waves PuigTec EQ» (емуляцію вінтажного еквалайзера) (додаток Д) для надання тембру теплоти. За допомогою еквалайзерів прибрано зайві низькочастотні призвуки та підкреслено бажані обертони, аби флейти звучали виразно і чисто [21].

Короткі різкі звуки струнних (піцикато, стакато) потребували спеціального підходу, щоб одночасно зберегти їх атаку і зробити звучання більш повним. На ці доріжки накладено компресор «FabFilter Pro-C 2» (додаток Е) із швидкими атакою та релізом. Швидка атака компресора миттєво приборкує різкий початковий транз'єнт кожної ноти, вирівнюючи його за рівнем із хвостом звучання [52]. У результаті відносна гучність сустейну підвищується: короткі ноти сприймаються більш протяжними та «теплішими», без різкого уколу на початку. Цей ефект підсилено додатковим transient shaper-ом: спеціальним процесором транз'єнтів відкориговано співвідношення атаки й сустейну. Зокрема, трохи зменшено надлишкову різкість атак і підкреслено тіло звуку, щоб піцикато не втрачали чіткості, але й не були надто колючими. В результаті короткі ноти струнних добре читаються в міксі і мають достатню гучність після початкового імпульсу, що особливо важливо в басу.

Щоб усі групи інструментів цілісно поєднались, застосовано техніку «Match EQ» з програмного пакету «iZotope Ozone» (додаток Ж). Цей інструмент

проаналізував частотний спектр міксу і порівняв його з еталонним, автоматично підлаштувавши еквалізацію під цільову криву [21]. Такий підхід допоміг згладити дисбаланси в тембрі і привести загальне звучання до єдиного стандарту. Фактично, «Match EQ» вирівняв тональний баланс, щоб жоден діапазон частот не домінував і не провалювався, роблячи звук більш цілісним. Це дозволило різним інструментам «сидіти» в міксі гармонійно, ніби вони з одного ансамблю, без відчуття розрізненості по тембру.

Важливою частиною звукорежисури оркестрового твору є створення відчуття спільного акустичного простору для всіх інструментів. Для цього використано два типи реверберації. Перш за все, за допомогою плагіна Soundtoys «SpaceBlender» (додаток К) додано художню реверберацію на окремі елементи. Це творчий алгоритмічний ревербератор, який дозволяє створювати нереальні, уявні простори і ефектні шари звучання [21]. «Space Blender» дав змогу отримати густі «хвилі» реверберації, що розгортаються у часі, додаючи кінематографічної масштабності мелодійним інструментам. Одночасно для всього оркестрового складу застосовано конволюційний ревербератор «Altiverb 7» (додаток Л) [8]. Він накладає імпульс від реального акустичного простору (концертної зали) на звук інструментів, створюючи ілюзію, ніби оркестр грає в реальному залі з природною акустиккою [21]. Всі інструменти були відправлені на один і той самий імпульс реверберації «Altiverb», завдяки чому склалося відчуття спільного просторового середовища, ніби оркестр знаходиться в одному приміщенні. Поєднання двох видів ревербераторів дало і баланс між креативним просторовим ефектом, і реалістичною «клеючою» реверберацією: «SpaceBlender» забезпечив експериментальні текстури й довгі шлейфи, а «Altiverb» - натуральну кімнатну «ауру» та глибину залу. Широке стерео-розміщення реверберації охопило увесь мікс, що також допомогло акустично об'єднати інструменти у єдиному просторі [52].

На завершальному етапі виконано мастеринг з використанням високоякісних процесорів, щоб надати треку комерційного блиску, збалансувати динаміку та підвищити гучність до необхідного рівня. Першим у мастер-

ланцюжку стояв компресор «Elysia Alpha Master» (додаток М), спеціалізований для мастерінгу. Він працює дуже прозоро і використаний для легкої «склейки» міксу: злегка компресує загальну динаміку стереоміксу, роблячи звучання більш щільним і цілісним, без помітних артефактів. Далі застосовано емуляцію аналогової стрічки - «Virtual Tape Machines» від Slate Digital [51]. Цей процесор насичує сигнал міксу теплою гармонійною дисторсією, імітуючи запис на магнітну стрічку, що додає глибини та «клею» між всіма елементами треку [21]. Стрічкова емуляція згладжує різкі піки та трохи ущільнює низькі частоти, роблячи загальне звучання більш насиченим та згуртованим.

Наступним кроком використано модуль Ozone Stabilizer - адаптивний еквалайзер з пакету iZotope Ozone, який в режимі реального часу аналізує спектр і автоматично підлаштовує частоти для більш гладкого тонального балансу. Stabilizer допоміг приборкати резонанси та виправити тональні перекося: якщо якась смуга частот випинала або провалювалася, модуль її вирівнював. У результаті частотна картинка стала ще рівнішою і приємнішою на слух. Додатково інженером внесено дрібні ручні правки еквалайзером на мастері - точкові підняття або послаблення окремих діапазонів для фінального полірування тембру [21].

Для кращого контролю над різними частотними областями застосовано мульти-смугову компресію. Мульти-смуговий компресор (multiband compressor) розділив сигнал на кілька діапазонів (низькі, середні, високі частоти) і стиснув кожен з них окремо. Наприклад, окремо компресовано низькі частоти, щоб бас став рівномірним і не занадто домінував; окремо оброблено середні, щоб, скажімо, згладити різкість смичкових; окремо високі - аби приборкати можливу різку цикоту від тарілок чи високих струнних. Такий підхід допоміг вирівняти енергію по всьому спектру, досягнувши збалансованого і одночасно щільного звучання треку. Мульти-смуговий компресор фактично забезпечив консистентний рівень гучності у різних частотних областях, що є ключовим для цілісності мастерінгу.

Останньою ланкою в мастеринг-ланцюжку стояв фінальний лімітер (із виходом на -1 dB). Лімітер підняв загальну гучність треку до таргетованого рівня -14 LUFS, обмежуючи пікові значення щоб уникнути цифрових перевантажень. Завдяки правильно налаштованому лімітеру вдалося досягти конкурентної гучності майстер-треку, не приносячи в жертву динамічний діапазон: піки були надійно «обрізані», але середній рівень звучання зріс без спотворень [24]. У підсумку фінальний мікс музичних фрагментів отримав професійне звучання: гучне, збалансоване по частотах і динамічно цілісне, готове до використання в ігровому проєкті або для демонстрації.

3.3. Технічна імплементація музики в ігровий рушій та налаштування параметрів адаптивності (FMOD, Unity)

Після створення адаптивного музичного контенту його необхідно інтегрувати в ігровий рушій та налаштувати реакцію на ігрові події. У даному проєкті для цього використовувався «аудіомідлвер FMOD Studio, інтегрований з рушієм «Unity». «FMOD» надає зручний інтерфейс для дизайну інтерактивної музики та прямі засоби з'єднання з «Unity» через плагін, що дозволяє прив'язати музичні події (Events) і параметри до логіки гри без написання значного обсягу коду [52]. Зокрема, «FMOD» підтримує обидва підходи до адаптивної музики - вертикальне нашарування та горизонтальне переключення / перекомпонування, а також керування параметрами в реальному часі для плавної зміни музики відповідно до стану гри [21].

У проєкті Unity було створено музичну подію в «FMOD Studio», що містить основні музичні треки та передбачає логіку переходів між ними. Цю подію імпортовано в «Unity» через офіційний «FMOD-плагін». У Unity використовується «Studio Event Emitter» або виклики «API FMOD» для відтворення музичної події та зміни її параметрів з коду гри. Таким чином, ігровий код відправляє в FMOD сигнали про зміну стану (наприклад, початок

рівня, виграш тощо) у вигляді зміни значення спеціального параметра. Параметри FMOD слугують для представлення стану гри і можуть динамічно змінювати звучання події. В даному випадку введено користувачький параметр (умовно названий «MusicState»), що набуває кількох дискретних значень, відповідних музичним станам гри. Логіка події в FMOD (маркери переходів, умови відтворення шарів тощо) налаштована так, щоб реагувати на зміну цього параметра і вмикати потрібний музичний розділ або шар.

Згідно задуму демо-гри було виділено три основні стани музики, між якими здійснюється перемикання. У фоновому режимі гри (очікування), коли гра перебуває в меню або гравець ще не розпочав ігровий процес, звучить фоновий варіант головної музичної теми. У цьому стані використовуються лише окремі інструментальні шари основної теми: залишено гармонічний та ритмічний шар, бас і головна мелодія вимкнені. Такий полегшений аранжування створює спокійну атмосферу і не перенавантажує увагу гравця. Технічно це реалізовано через вертикальне нашарування: на таймлайні FMOD основна тема розкладена на декілька треків (шарів), і для фонового режиму гучність окремих треків (басу, мелодії) знижена до нуля або ці треки відпочатково вимкнені. Параметр «MusicState = 0» (умовно) відповідає цьому стану і через «Automation» або умовні логічні маркери забезпечує відтворення тільки потрібних шарів (додаток Н). В результаті при знаходженні гри у фоновому режимі подія грає цикл основної теми без насичених елементів, підтримуючи легкий фон.

У основному рівні, коли гравець починає рівень, музика переходить у повноцінну головну тему. В цьому стані задіяна повна версія основного треку з усіма шарами - додаються раніше приглушені басова лінія та мелодія, що підсилює динаміку. Важливо, що перехід від фонового варіанту до повної теми відбувається без переривання ритму та темпу музики. Це досягнуто завдяки тому, що обидва варіанти є частинами однієї події «FMOD»: як тільки гравець стартує рівень (параметр «MusicState» змінюється, скажімо, на 1), «FMOD» або вмикає додаткові треки у поточному такті, або здійснює перехід на іншу секцію таймлайну з відносною прив'язкою позиції [38]. Був обраний підхід з

використанням переключення секцій (горизонтальний підхід) із збереженням позиції відтворення, оскільки він простіше у налаштуванні та гарантує синхронність введення нових шарів [**Ошибка! Источник ссылки не найден.**]. У результаті, коли гра переходить у активну фазу, слухач плавно, непомітно для себе, чує розгортання повного звучання теми.

Третій стан передбачає ситуацію, коли в грі спрацьовує певний тригер підвищення напруги (запущено таймер). Музично це вирішено через переключення на іншу тему - додатковий трек, який відрізняється від основного і посилює відчуття напруженості. Перехід на другу тему здійснюється методами горизонтального ресеквенування і супроводжується спеціальними налаштуваннями для плавності. Зокрема, в FMOD для відповідного маркеру переходу встановлено квантування 1 такт (One bar) [38]. Це означає, що після зміни параметра (наприклад, MusicState з 1 на 2), система не перемикає музику миттєво, а чекає найближчої межі такту перед тим, як активувати новий музичний розділ. Такий підхід гарантує, що перехід відбудеться чітко в ритмі і початок другого треку співпаде з початком тактової сітки першого треку.

У нашій реалізації основна тема продовжує грати до кінця поточного такту, після чого замість наступного циклу основної теми за вступним ударом такту запускається друга тема. Завдяки тому, що обидві музичні теми узгоджені по тональності та розміру, такий перехід сприймається природно. Ба більше, використано невеликий кросфейд і збіг музичних акцентів, тож зміна настрою відбувається плавно, без різких обривів. Подібні техніки широко застосовуються у сучасних іграх: наприклад, у «Hades» (Supergiant Games) під час бою FMOD перемикається на більш інтенсивну частину саундтреку і додає нові «важкі» інструментальні шари, причому завершення бою синхронізується із закінченням музичної фрази [38]. У нашому випадку відквантований перехід на другий трек дозволив досягти аналогічної плавності та виразності музичної реакції на події гри.

Для керування зазначеними переходами використано єдиний параметр MusicState (лейблового типу) з трьома станами, що змінюється скриптом Unity

при настанні відповідних подій гри. «FMOD Studio» інтерпретує ці зміни параметра згідно закладеної логіки: активує або деактивує потрібні інструментальні треки, або перемикає таймлайн на інший розділ з врахуванням зазначеного квантування переходів. У результаті вдалося реалізувати адаптивний музичний супровід, що складається з кількох рівнів інтенсивності та безперервно підлаштовується до ходу ігрового процесу. Така технічна імплементація забезпечила синхронність аудіо з дією і мінімізувала ризик помітних для гравця розривів в музиці, оскільки всі зміни відбуваються точно в такт та під контролем аудіосистеми FMOD [39].

Джерела Unity-скриптів опитують стан гри (події старту рівня, закінчення таймеру тощо) і встановлюють відповідне значення параметра в FMOD (через функції «set Parameter By ID» або подібні засоби API), тим самим запускаючи потрібну музичну реакцію. Така декомпозиція логіки (гра формує події, а аудіомідлвейр самостійно реалізує сценарій переходів) спрощує налаштування та відлагодження саундтреку і дозволяє композитору наперед продумати та протестувати всі музичні переходи у середовищі «FMOD Studio», використовуючи «Live Update» для перевірки в грі [39]. Таким чином, технічна імплементація адаптивної музики на базі зв'язки «Unity + FMOD» дала можливість досягти поставленої художньої мети - зробити музичне оформлення гнучким, інтерактивним і водночас цілісним за звучанням у межах демо-проєкту.

У межах тієї ж аудіосистеми було реалізовано й стінгери виграшу та програшу, які доповнюють основну адаптивну музику короткими реактивними музичними сигналами. Для цього в «FMOD Studio» створено окрему подію, призначену спеціально для відтворення стінгерів, у структурі якої передбачено дві незалежні варіації - виграшну та програшну. Обидві варіації зберігаються всередині «Event-a» під назвою «Level_End» у вигляді двох мультішот-треків. Така організація дозволила зберегти всі короткі реакційні звуки в одному логічному модулі та спростила керування ними з боку Unity.

Усередині цього «Event-a» запроваджено окремий параметр стінгера (умовно «StingerType»), що має два дискретні значення, відповідні результату

ігрової події. Принцип роботи аналогічний структурі «MusicState»: зміна значення параметра автоматично активує відтворення потрібної варіації всередині тієї ж FMOD-події. Логіка події в FMOD побудована так, що у разі вибору варіації «виграш» запускається яскрава мажорна фраза з висхідним рухом, а у разі вибору «програш» - короткий мінорний сигнал з характерним нисхідним жестом. Варіації реалізовані як однократні, завершені музичні фрази, які не потребують циклічного відтворення й автоматично зупиняються після завершення власного таймлайна.

У «Unity» цей «Event» підключено аналогічно до основної музичної події - через «Studio Event Emitter» або методи API. Ігровий код викликає стінгер не за допомогою зміни музичного параметра, а через окремий програмний тригер (наприклад, у момент завершення рівня). У відповідній частині коду виконується встановлення параметра «StingerType» у значення «виграш» або «програш», після чого здійснюється одноразовий виклик «Play» для цієї події. Важливо, що сам «Event» не перебуває в постійному циклі, як основна музика, а запускається лише за потреби, що повністю відповідає природі стінгерів. Така схема дозволяє уникнути зайвого навантаження на систему й гарантує, що реактивні музичні сигнали будуть відтворюватися лише у відповідні моменти ігрового процесу.

Завдяки тому, що всі параметри логіки стінгерів розміщено безпосередньо в FMOD, була змога протестувати та налаштувати їх поведінку без необхідності постійно змінювати ігровий код. «Live Update» у «FMOD Studio» дозволив перевірити звучання стінгерів у реальному часі. Таким чином, впровадження стінгерів у «Gem Hunter» (додаток 1) стало органічним продовженням загальної концепції адаптивної музики: короткі виграшні та програшні фрази відтворюються точно в потрібний момент, не порушуючи ритмічної та тембрової цілісності саундтреку й підсилюючи емоційну реакцію гравця на ключові події гри.

ВИСНОВКИ

Історико-теоретичний аналіз розвитку ігрової музики підтвердив еволюційний характер становлення адаптивних саундтреків. Визначено, що основи інтерактивної музики виникли ще за часів 8-бітних ігор як відповідь на технічні ліміти: циклічність, модульність та варіативність стали характерними рисами ігрових тем. Подальший розвиток (1990–2000-ті рр.) привів до ускладнення саундтреків, впровадження стилістичних впливів мінімалізму, електронної та ембієнт-музики, а також до появи перших прикладів адаптивного аудіо в іграх. На початку 21-го століття із зростанням обчислювальних можливостей ігрові саундтреки еволюціонували в повноцінні оркестрові партитури з інтерактивними елементами. Таким чином, сучасна адаптивна музика в іграх постала як синтез багаторічної композиторської практики та новітніх технологій.

Систематизація сучасних композиційних підходів показала, що в основі адаптивних саундтреків лежать два базові методи: вертикальний лєєринг (динамічне додавання/прибирання шарів) та горизонтальне ресеквенсування (зміна послідовності або варіантів музичних фрагментів). Обидва методи забезпечують реальні варіації музики у відповідь на події гри, але різними способами. Перший відповідає за текстурну змінність (нарощування чи зменшення інструментування без переривання музики), другий – за структурну гнучкість (стрибки між різними тематичними розділами саундтреку). У практиці вони часто комбінуються, створюючи гібридні схеми адаптації. Також виявлено допоміжні техніки, зокрема параметричні зміни (керування гучністю, темпом, фільтрацією в реальному часі) та стінгери (короткі музичні вставки), які підсилюють адаптивність музики. Проаналізовано приклади застосування цих підходів у сучасних іграх (динамічний саундтрек «DOOM» (2016) на основі модульного лєєрингу, варіативна гармонія в «Halo 2», горизонтально-вертикальний гібрид у «Untitled Goose Game» тощо).

Вивчення технологічних рішень засвідчило, що для реалізації адаптивної музики на практиці використовуються як можливості ігрових рушіїв (вбудовані аудіосистеми Unity та Unreal Engine), так і спеціалізовані «аудіо-мідлвеари» (FMOD, Wwise). Проведений порівняльний аналіз показав, що Unreal Engine володіє потужним нативним аудіомодулем, тоді як Unity потребує більшого залучення сторонніх інструментів для складного аудіо. Серед мідлвеарів *Wwise* доцільно застосовувати у великих проектах з розгалуженою аудіосистемою, а FMOD – у менш масштабних або інди-проектах завдяки його простоті та безкоштовній ліцензії. Обидва інструменти забезпечують усі необхідні функції для адаптивної музики (підтримка багатошаровості, параметричного мікшування, синхронізації з подіями гри та ін.). Визначено, що оптимальною стратегією є інтеграція аудіомідлвеару з рушієм гри: це розширює можливості саунд-дизайнерів і спрощує процес імплементації.

За результатами дослідження було розроблено концепцію адаптивного музичного оформлення для демо-проєкту «Gem Hunter Match». На основі аналізу жанрових вимог (казуальна головоломка) та технічних можливостей рушія Unity і мідлвейру FMOD було запропоновано гібридну схему саундтреку, що поєднує фонову циклічну тему з додатковими шарами інтенсивності та окремими музичними сигналами на події. Сформовано художній задум саундтреку: музика мала бути легкою, ненав'язливою в основі, але реагувати на ігрові дії переходом до відповідних музичних фрагментів та нашаруванням стемів. Обґрунтовано використання саме оркестрового жанру для досягнення необхідного балансу між фоновістю і виразністю.

Здійснено практичну реалізацію концепції за допомогою зв'язки Unity + FMOD. Створено оригінальний музичний контент (дві основні теми) згідно із задумом. За допомогою FMOD Studio розроблено логіку адаптивного аранжування: налаштовано параметри переключення між темами, точки синхронізації переходів, гучність шарів тощо. Далі інтерактивну аудіосистему інтегровано в проєкт Unity через офіційний плагін FMOD. Програмну реалізацію здійснено скриптами C#, які відстежують стан гри (початок рівня, закінчення

часу) та змінюють значення параметра інтенсивності музики в FMOD. У результаті музика в демо-грі почала динамічно підлаштовуватися до перебігу ігрового процесу.

Якісний аналіз результатів показав ефективність застосованих композиційних і технологічних методів. Створений адаптивний саундтрек забезпечує необхідну гнучкість: музика безперервно супроводжує ігровий процес і в той же час змінюється залежно від дій гравця. Застосування вертикального та горизонтального методів аранжування дало змогу досягти балансу між варіативністю та цілісністю: саундтрек сприймається як єдине ціле, але не повторюється дослівно і не набридає при тривалому прослуховуванні. Технічна імплементація на базі Unity та FMOD продемонструвала свою надійність і зручність: всі музичні події коректно синхронізуються з ігровими, а система легко масштабується для потенційного розширення (можна додати нові рівні, теми чи стінгери без зміни вже написаного коду гри).

Загалом, результати дослідження підтверджують, що адаптивна музика у відеоіграх є ефективним засобом підвищення рівня занурення і емоційного впливу на гравця. Поставлені у вступі завдання виконано: історичну динаміку розвитку ігрової музики простежено, поняття та методи адаптивної музики розкрито, практичні приклади проаналізовано, а сучасні технології аудіоінтеграції узагальнено. Досягнення мети роботи – всебічного аналізу явища адаптивного музичного супроводу – дозволяє окреслити перспективи подальших досліджень. Зокрема, наступними кроками можуть стати глибше вивчення впливу динамічної музики на поведінку гравців, експерименти з алгоритмічною генерацією музики за допомогою штучного інтелекту, а також розробка рекомендацій для композиторів щодо оптимального балансу інтерактивності та художньої цілісності саундтреку. Завдяки поєднанню теоретичного підходу і аналізу практичних кейсів, ця робота робить внесок у розуміння того, як музика здатна адаптуватися до інтерактивного медіа і збагачувати досвід гравця.

СПИСОК ВИКОРИСТАНИХ ДЖЕРЕЛ

1. Адаптивна музика в слотах: як звук створює атмосферу перемоги. *Мій фронт* [Електронний ресурс]. URL: <https://myfront.in.ua/adaptyvna-muzyka-v-slotakh-iaak-zvuk-stvoriuie-atmosferu-peremohy/> (дата звернення: 10.09.2025).
2. Вагерич В. В. Значення звуку та музики у відеоіграх. *Культурологічний альманах*. 2024. № 1. С. 273–277. DOI: <https://doi.org/10.31392/cult.alm.2024.1.34>.
3. Пожарицька О. Візуальне мистецтво та відеогра як витoki нових жанрів дигітальної поезії. *Лінгвістичні студії*. 2021. № 41. С. 225–234. DOI: <https://doi.org/10.31558/1815-3070.2021.41.22>.
4. Папченко В. П. Інтерактивність шумів і пауз у звуковому образі фільму. *Науковий вісник Київського національного університету театру, кіно і телебачення імені І. К. Карпенка-Карого*. 2018. № 22. С. 101–107. DOI: <https://doi.org/10.34026/1997-4264.22.2018.217057>.
5. Хренов Д. О. Перспективи розвитку саунд-дизайну в Україні. *Вісник КНУКіМ. Серія «Мистецтвознавство»*. 2019. № 40. С. 18–26. DOI: <https://doi.org/10.31866/2410-1176.40.2019.173960>.
6. A Landmark Timeline of Video Game Music / M. Fritsch, T. Summers (eds.). *The Cambridge Companion to Video Game Music*. Cambridge : Cambridge University Press, 2021. Р. xxii–xxxii. DOI: <https://doi.org/10.1017/9781108670289>.
7. Adam N. Audio Integration in Unreal Engine Games. *SDLC Corp.* [Електронний ресурс]. 2025. URL: <https://sdllcorp.com/post/sound-design-and-audio-integration-in-unreal-engine-games-2/> (дата звернення: 21.10.2025).
8. Adaptive music in video games: What it is and how it works. *Blips.fm* [Електронний ресурс]. URL: <https://blog.blips.fm/articles/adaptive-music-in-video-games-what-it-is-and-how-it-works> (дата звернення: 21.09.2025).
9. *Altiverb 8*: імпульсний ревербератор. AudioEase [Електронний ресурс]. URL: <https://www.audioease.com/altiverb> (дата звернення: 20.09.2025).
10. *Blips.fm*: Adaptive music software: A roundup of the best options for video games. [Електронний ресурс]. URL: <https://blog.blips.fm/> (дата звернення: 16.09.2025).
11. Bosso J. Meet the 'Red Dead Redemption II' Guitar Slinger, Woody Jackson. *Guitar World* [Електронний ресурс]. 2019. URL: <https://www.guitarworld.com/artists/meet-the-read-dead-redemption-ii-guitar-slinger-woody-jackson> (дата звернення: 10.10.2025).

12. Chord Progressions That Work Every Time. *Octaton* [Электронный ресурс]. URL: <https://www.octaton.com/blog/chord-progressions-that-work-every-time> (дата звернения: 05.10.2025).
13. Collins K. *Game Sound: An Introduction to the History, Theory, and Practice of Video Game Music and Sound Design*. Cambridge, MA : The MIT Press, 2008. 216 p.
14. Collins K. In the Loop: Creativity and Constraint in 8-bit Video Game Audio. *Twentieth-Century Music*. 2007. Vol. 4, № 2. P. 209–227. DOI: 10.1017/S1478572208000510.
15. DeFilippis J. M. Mastering and distributing immersive sound. *Immersive Sound Production*. London : Routledge, 2022. P. 123–140. DOI: <https://doi.org/10.4324/9781003052876-8>
16. Dídac. Multiband Compressor in Mastering: When, Why & How. *MasteringBOX* [Электронный ресурс]. 2023. URL: <https://www.masteringbox.com/learn/multiband-compression> (дата звернения: 11.11.2025).
17. Donahoe B. The History of Video Game Music: Signal Cannon Podcast. *Press For Sound*, 2017. URL: <https://pressforsound.com/2017/10/27/the-history-of-video-game-music/> (дата звернения: 19.09.2025)
18. Donnelly K. J., Gibbons W., Lerner N. *Music in Video Games: Studying Play*. New York; London : Routledge, 2014. 246 p.
19. Elysia Alpha Compressor (User Manual). URL: https://files.plugin-alliance.com/products/elysia_alpha_compressor/elysia_alpha_compressor_manual.pdf (дата звернения: 20.10.2025).
20. Fritsch M., Summers T. *The Cambridge Companion to Video Game Music*. Cambridge : Cambridge University Press, 2021. 484 p. DOI: 10.1017/9781108670289.
21. Game Audio Development: Complete Guide to Sound Design, Music & Implementation. *Generalist Programmer*, October 3, 2025 URL: <https://generalistprogrammer.com/game-audio-development/> (дата звернения: 11.10.2025).
22. Game Music Synchronisation: A Guide to Dynamic Audio Implementation *Twine Blog*. URL: <https://www.twine.net/blog/game-music-synchronisation-a-guide-to-dynamic-audio-implementation/> (дата звернения: 29.09.2025).
23. Geluso P. Stereo / *Immersive sound*. New York ; London : Routledge, 2017. P. 63–87. DOI: <https://doi.org/10.4324/9781315707525-4>.
24. Get More Out Of Your Compressor With Attack Settings. *Recording Revolution*, Dec 16, 2011. URL: <https://www.recordingrevolution.com/blog/get-more-out-of-your-compressor-with-attack-settings> (дата звернения: 26.09.2025).

25. Hoff J. Music Design of Assassin's Creed Shadows. *Audiokinetic*. 2020. URL: <https://www.audiokinetic.com/en/blog/music-design-assassins-creed-shadows/> (дата звернення: 29.09.2025)
26. Jones J. Brian Eno Explains the Origins of Ambient Music. *OpenCulture*. 2021. URL: <https://www.openculture.com/2021/03/brian-eno-explains-the-origins-of-ambient-music.html> (дата звернення: 15.10.2025)
27. Mack H. Atmospheric vs Ambient Music in Games. *Game Developer*, 2015. URL: <https://www.gamedeveloper.com/audio/getting-making-game-music-that-fits---comparative-music-series---atmospheric-vs-ambient/> (дата звернення: 12.12.2025).
28. Mack. H. Atmospheric vs Ambient Music in Games. *Gamedeveloper.com*. 2015. URL: <https://www.gamedeveloper.com/audio/getting-making-game-music-that-fits---comparative-music-series---atmospheric-vs-ambient/> (дата звернення: 21.10.2025)
29. Montandon L. *Creating an adaptive music system for the video game prototype Henry*: Master's thesis. Zürcher Hochschule der Künste, 2022. 57 p. URL: <https://www.zhdk.ch/file/live/16/1624f9f2a6a3581bf91aa0a964e8d90b429609e6/1ucienmontandonmasterthesis.pdf> (дата звернення: 12.10.2025).
30. Ombler M. Megadrive to Mega Hit: Why Video Games Are So Tied to Club Music. *VICE*, 29.10.2018. URL: <https://www.vice.com/en/article/video-games-90s-club-music-commodore-amiga/> (дата звернення: 30.09.2025).
31. Parkinson J. The Sound of Hades. *Gameplay* Dec 24, 2020 URL: <https://gameplay.co/hades-game-music-sound-design-darren-korb-supergiant-games> (дата звернення: 17.10.2025).
32. Phillips W. *A Composer's Guide to Game Music*. Cambridge, MA : MIT Press, 2014. 275 p.
33. Phillips W. Horizontal Resequencing and Song Structure for Game Music Composers: From Spyder to Sackboy (GDC 2021). [Електронний ресурс]. 2021. URL: <https://winifredphillips.wpcomstaging.com/2021/08/15/horizontal-resequencing-and-song-structure-for-game-music-composers-from-spyder-to-sackboy-gdc-2021/> (дата звернення: 18.10.2025).
34. Proschaylo A. Secrets of DOOM's Music: Mick Gordon's Adaptive Soundtrack and Its Impact on Gameplay. *EnsiGame*, 2021. URL: <https://ensigame.com/articles/gaming/secrets-of-dooms-music-mick-gordons-adaptive-soundtrack-and-its-impact-on-gameplay-2> (дата звернення: 12.10.2025).
35. Reichert T. Mycorrhiza's Dynamic Audio. *Devlog*, 2022. URL: <https://fulminisictus.itch.io/mycorrhiza/devlog/401340/mycorrhizas-dynamic-audio/> (дата звернення: 10.11.2025).

36. Sanders T. A., Cairns P. Time perception, immersion and music in videogames. Proceedings of HCI 2010. 2010. URL: <https://doi.org/10.14236/ewic/hci2010.21> (дата звернення: 30.09.2025).
37. Schartmann A. Koji Kondo's Super Mario Bros. Soundtrack. New York; London : Bloomsbury Academic, 2015. 146 p.
38. Seamless Transitions for Complicated Adaptive Music Systems. *FMOD Forum*. URL: <https://qa.fmod.com/t/seamless-transitions-for-complicated-adaptive-music-systems/19480/> (дата звернення: 26.09.2025).
39. Setting FMOD Parameters in Unity. *FMOD*. URL: <https://alessandrofama.com/tutorials/fmod/unity/parameters/> (дата звернення: 26.09.2025).
40. Singer N. Halo, and the birth of cinematic adaptive music. URL: <https://www.nicholassinger.com/blog/halo-and-the-birth-of-cinematic-adaptive-music/> (дата звернення: 17.10.2025).
41. Sound Engine: FMOD. *Mobygames* URL: <https://www.mobygames.com/group/13947/sound-engine-fmod> (дата звернення: 10.09.2025).
42. Space Blender. Product Page. *Soundtoys*. URL: <https://www.soundtoys.com/product/spaceblender/> (дата звернення: 20.10.2025).
43. Summers T. Music and the Video Game as Ritual Encounter. URL: https://pure.royalholloway.ac.uk/ws/portalfiles/portal/36443297/Naxos_Games_Music_Ritual_Final.pdf (дата звернення: 12.10.2025).
44. Summers T. Understanding Video Game Music. Cambridge : Cambridge University Press, 2016. 262 p. DOI:10.1017/CBO9781316337851
45. Sweet M. Writing interactive music for video games : a composer's guide. Boston : Addison-Wesley, 2014. 510 p.
46. Szafraniec J. Where did game music come from — minimalism, pt. 1. *Gamemusic.net*. 2019. URL: <https://gamemusic.net/game-music-minimalism-1/> (дата звернення: 15.10.2025).
47. The Evolution of Video Game Music: From 8-Bit to Orchestral Scores. *ELVTR UK*. URL: <https://uk.elvtr.com/blog/the-evolution-of-video-game-music/> (дата звернення: 10.09.2025)
48. The Role of Adaptive Music in Creating Immersive Game Worlds. *The Game Audio Company*, 2023. URL: <https://www.thegameaudioco.com/the-role-of-adaptive-music-in-creating-immersive-game-worlds> (дата звернення: 16.09.2025).
49. The Ultimate Guide to Reverb. *Get That Pro Sound*. URL: <https://getthatprosound.com/ultimate-guides/reverb/> (дата звернення: 17.10.2025).

50. Understanding Audio Implementation in Unreal Engine and Unity. *The Game Audio Company*, 2025. URL: <https://www.thegameaudioco.com/understanding-audio-implementation-in-unreal-engine-and-unity> (дата звернення: 16.09.2025).
51. Virtual Tape Machines. Product Page. *Slate Digital*. URL: <https://slatedigital.com/virtual-tape-machines/> (дата звернення: 20.10.2025).
52. Weir P. Stealing Sound: The Application of Generative Music. Game Developers Conference (GDC 2011). *GDC Vault*. URL: <https://www.gdcvault.com/play/1014482/Stealing-Sound-The-Application-of-Generative-Music> (дата звернення: 12.10.2025).
53. What Makes a Great RPG Soundtrack? A Composer's Perspective. *Ouranio Recordings*. URL: <https://ouraniorcordings.com/what-makes-a-great-rpg-soundtrack-a-composers-perspective/> (дата звернення: 11.10.2025)
54. Wwise or FMOD? A Guide to Choosing the Right Audio Tool for Every Game Developer. *The Game Audio Company*, 2024. URL: <https://www.thegameaudioco.com/wwise-or-fmod-a-guide-to-choosing-the-right-audio-tool-for-every-game-developer> (дата звернення: 16.09.2025).
55. Wwise, the most advanced, feature-rich interactive audio solution. *Audiokinetic*. URL: <https://www.audiokinetic.com/en/wwise/overview> (дата звернення: 19.10.2025)
56. Yarwood J. Playing Your Song: The Evolution of Dynamic Music in Games *EGM*. 2019. URL: <https://egmnow.com/playing-your-song-the-evolution-of-dynamic-music-in-games/> (дата звернення: 30.10.2025).

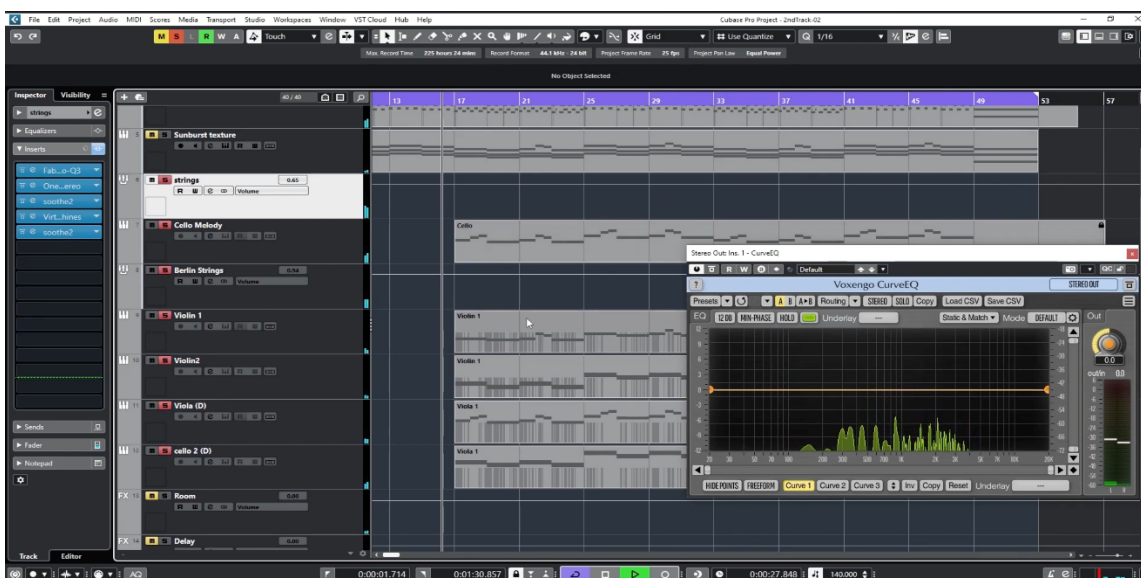
ДОДАТКИ

ДОДАТОК А

Носій з аудіофайлами авторських композицій для гри «Gem Hunter» та демо-збіркою гри «Gem Hunter».

ДОДАТОК Б

Скріншот з проєкту в Steinberg Cubase.



ДОДАТОК В

Скріншот плагіну CLA-2A від Waves.



ДОДАТОК Г

Скріншот плагіну Fabfilter Pro-Q 4.



ДОДАТОК Д

Скріншот плагіну Puig Tec EQ від Waves.



ДОДАТОК Е.

Скріншот плагіну Fabfilter Pro-C 2.



ДОДАТОК Ж

Скріншот плагіну Izotope Ozone Match EQ.



ДОДАТОК К

Скріншот плагіну Soundtoys SpaceBlender.



ДОДАТОК Л

Скріншот плагіну Altiverb by AudioEase.



ДОДАТОК М

Скріншот плагіну Elysia Alpha Master.



ДОДАТОК Н.

Скріншот з проекту FMOD.

