

МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ
ЗАПОРІЗЬКИЙ НАЦІОНАЛЬНИЙ УНІВЕРСИТЕТ

МАТЕМАТИЧНИЙ ФАКУЛЬТЕТ

Кафедра комп'ютерних наук

КВАЛІФІКАЦІЙНА РОБОТА МАГІСТРА

на тему: «РОЗРОБКА НАВЧАЛЬНОЇ КОМП'ЮТЕРНОЇ
ПРОГРАМИ ДЛЯ ВИКОРИСТАННЯ НА УРОКАХ З
МУЗИКИ»

Виконала: студентка 2 курсу, групи 8.1220-з
спеціальності 122 комп'ютерні науки
(шифр і назва спеціальності)

освітньої програми комп'ютерні науки
(назва освітньої програми)

С. Б. Зайцева

(ініціали та прізвище)

Керівник доцент кафедри комп'ютерних наук,
доцент, к.т.н. Решевська К.С.
(посада, вчене звання, науковий ступінь, прізвище та ініціали)

Рецензент завідувач кафедри програмної інженерії
доцент, к.ф.-м.н. А.О. Лісняк.
(посада, вчене звання, науковий ступінь, прізвище та ініціали)

МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ
ЗАПОРІЗЬКИЙ НАЦІОНАЛЬНИЙ УНІВЕРСИТЕТ

Факультет математичний
Кафедра комп'ютерних наук
Рівень вищої освіти магістр
Спеціальність 122 комп'ютерні науки
(шифр і назва)
Освітня програма комп'ютерні науки

ЗАТВЕРДЖУЮ

Завідувач кафедри комп'ютерних наук, к.т.н., доцент

_____ Борю С.Ю.
(підпис)

« 14 » _____ 06 2021 р.

З А В Д А Н Н Я
НА КВАЛІФІКАЦІЙНУ РОБОТУ СТУДЕНТОВІ

Зайцевій Світлані Борисівні

(прізвище, ім'я та по-батькові)

1. Тема роботи (проекту) Розробка навчальної комп'ютерної програми для використання на уроках з музики

Решевська Катерина Сергіївна, к.т.н., доцент кафедри комп'ютерних наук

керівник роботи (проекту) _____
(прізвище, ім'я та по-батькові, науковий ступінь, вчене звання)

затверджені наказом ЗНУ від « 09 » _____ червня 2021 року № 850-с

2. Строк подання студентом роботи 19.11.2021

3. Вихідні дані до роботи 1. Постановка задачі
2. Перелік літератури

4. Зміст розрахунково-пояснювальної записки (перелік питань, які потрібно розробити)
1. Постановка задачі
2. Основні теоретичні відомості
3. Розробка навчальної комп'ютерної програми

5. Перелік графічного матеріалу (з точним зазначенням обов'язкових креслень) _____
презентація

6. Консультанти розділів роботи

Розділ	Прізвище, ініціали та посада консультанта	Підпис, дата	
		завдання видав	завдання прийняв

7. Дата видачі завдання _____ 14.06.2021 _____

КАЛЕНДАРНИЙ ПЛАН

№	Назва етапів кваліфікаційної роботи магістра	Строк виконання етапів роботи	Примітка
1.	Розробка плану роботи.	25.06.2021	
2.	Збір вихідних даних.	09.09.2021	
3.	Обробка методичних та теоретичних джерел.	03.10.2021	
4.	Розробка першого і другого розділу.	16.10.2021	
5.	Розробка третього розділу.	29.10.2021	
6.	Оформлення та нормоконтроль кваліфікаційної роботи	26.11.2021	
7.	Захист кваліфікаційної роботи	10.02.2022	

Студент _____
(підпис)

С. Б. Зайцева _____
(ініціали та прізвище)

Керівник роботи _____
(підпис)

К. С. Решевська _____
(ініціали та прізвище)

Нормоконтроль пройдено

Нормоконтролер _____
(підпис)

О. Г. Спиця _____
(ініціали та прізвище)

РЕФЕРАТ

Кваліфікаційна робота магістра «Розробка навчальної комп'ютерної програми для використання на уроках з музики»: 65 с., 32 рис., 29 джерел.

ЗВУКОВИЙ АНАЛІЗ, МУЗИКА, НАВЧАЛЬНА КОМП'ЮТЕРНА ПРОГРАМА, ОНЛАЙН-НАВЧАННЯ, DJANGO, PYTHON, SQLITE.

Об'єкт дослідження – використання навчальної комп'ютерної програми для вивчення предмету музичне мистецтво у середній школі.

Мета роботи – покращити процес навчання з предмету музика шляхом використання ІТ-технологій, а саме створення індивідуальної навчальної програми.

Методи дослідження – аналіз, синтез, класифікація, спостереження, порівняння, узагальнення.

У кваліфікаційній роботі власний веб-сервіс розглядається як освітній інструмент для навчання учнів у школах та вищих навчальних закладах, для проведення науково-педагогічних досліджень шляхом впровадження елементів гейміфікації на уроках, а саме лекцій із практичними завданнями, тестів і модулю для аналізу звуку за допомогою аналізатора аудіоспектру. Впровадження цих елементів повинно спростити процес ведення уроків, особливо під час дистанційного навчання, та зацікавити учнів інформаційними технологіями та їх впливом на усі галузі.

На основі платформи для розробки веб-додатків Python Django створена система управління навчанням, що включає інструментарій для проведення онлайн-навчання, можливість створювати акаунти користувачів, групи (або класи), також можливість створювати уроки та тести з налаштовуваним контентом та використовувати індивідуальний модуль, що впроваджує аналізатор аудіоспектру для аналізу та обробки звуку на прикладі завантажених на сервер необроблених аудіозаписів.

SUMMARY

Master's Qualification Thesis "Development of Educational Music Software": 65 pages, 32 pictures, 29 references.

SOUND ANALYSIS, MUSIC, LEARNING MANAGEMENT SYSTEM, ONLINE EDUCATION, DJANGO, PYTHON, SQLITE.

The object of research is the use of a learning management system for studying the subject of art in secondary school.

The Aim of the work is to improve the process of learning the subject of music through the use of IT technologies, namely the creation of an individual course program.

Research methods – analysis, synthesis, classification, observation, comparison, generalization.

In the qualification work, the own-made web service is considered as educational tool for teaching students at schools and universities, for conducting scientific and pedagogical research by introducing elements of gamification in the classroom, namely lectures with practical tasks, tests and a module for sound analysis using an audio spectrum analyzer. The introduction of these elements should simplify the process of conducting lessons, especially during distance learning, and interest students in information technology and their impact on all areas.

Based on the Python Django web application development platform, a learning management system has been created, which includes tools for online learning, as well as the ability to create user accounts, groups (or classes), as well as the ability to create lessons and tests with customizable content and use individual module, which implements an audio spectrum analyzer for sound analysis and processing on the example of unprocessed audio recordings uploaded to the server.

ЗМІСТ

Завдання на кваліфікаційну роботу.....	2
Реферат	4
Summary	5
Скорочення та умовні позначки	8
Вступ.....	9
1 Теоретичні аспекти комп'ютерних навчальних програм та способів їх використання	12
1.1 Поняття комп'ютерної навчальної програми.....	12
1.2 Види комп'ютерних навчальних програм	13
1.2.1 Електронні курси.....	14
1.2.2 Помічники для класів	14
1.2.3 ПЗ для оцінювання показників успішності	15
1.2.4 Електронні довідники	16
1.2.5 Індивідуальні програми	16
1.3 Існуючі рішення, їх особливості, переваги та недоліки.....	17
1.3.1 Особливості системи Moodle	17
1.3.2 Особливості системи Blackboard Learn	21
1.3.3 Особливості системи Canvas.....	23
1.4 Висновки за розділом 1	25
2 Аналіз, технологічне та алгоритмічне забезпечення програмного рішення.....	27
2.1 Сутність проблеми звукового аналізу.....	27
2.1.1 Аналізатор спектру та швидке перетворення Фур'є	28
2.1.2 Візуальне представлення спектру аудіохвиль	31
2.2 Методи наукового дослідження виконаної роботи	32
2.3 Специфікація вимог програмного забезпечення	35
2.3.1 Призначення	35

2.3.2 Аудиторія та потенційні користувачі продукту	35
2.3.3 Обсяг проекту	35
2.3.4 Характеристики продукту	36
2.3.5 Вимоги до розробки продукту	40
2.3.6 Архітектура програмного засобу.....	41
2.3.7 Використані технології.....	42
2.4 Висновки за розділом 2	43
3 Розробка навчальної програми для використання на уроках музики.....	45
3.1 Загальний вигляд та функціонал розробленого веб-додатку	45
3.2 Тестування розробленої програми	57
3.2 Висновки за розділом 3	60
Висновки	62
Перелік посилань.....	63

СКОРОЧЕННЯ ТА УМОВНІ ПОЗНАКИ

БД – база даних

СУБД – система управління базами даних

ER – Entity-Relationship (модель «сутність-зв'язок»)

RDM – реляційна модель даних

UML – діаграма класів

ПЗ – програмний засіб

VLE – віртуальне середовище навчання

LMS – система управління навчанням

ШПФ – швидке перетворення Фур'є

РЧ – радіочастота

STFT – Short-Time Fourier Transform

SMS – Spectral Modeling Synthesis

ВСТУП

Ми усі добре знаємо про те, як люди та тварини сприймають звук. Але як це робить машина? Як це робить електронний музичний інструмент? Це питання, на які спочатку неможливо відповісти. Можна міркувати годинами і сказати, що машини добре лічать і тому швидко перетворюють цифри на звуки і навпаки, і це буде, мабуть, найточнішим описом того, що відбувається в машинних інструкціях, або, як то кажуть, в голові у машини, якщо відповідати навмання. Проте, на сьогодні роль вчителя в тому і полягає, щоб пояснити учням, як відбувається ця взаємодія між машинним та природним світами на практиці та як зацікавити людей своїм предметом, підштовхнути їх до його вивчення шляхом впровадження нових технологій.

Звичайно, робота вчителя у цьому сторіччі трохи змінилася, оскільки традиційні методи навчання залишилися у минулому. Вчитель більше не є основним джерелом інформації – він виступає скоріше у ролі гіда у світ знань. А так як прогрес не стоїть на місці, то кожен вчитель користується інформаційними технологіями для можливості ділитися з учнями найбільш повним та обширним за обсягом інструментарієм та багажем знань.

Від багатьох учителів на сьогодні також вимагають беззаперечної комп'ютерної грамотності та підштовхують їх до ведення блогів або персональних вебсайтів – та це вже сприймається як даність, тому що всюди активно іде діджиталізація [1]. А створення індивідуального програмного засобу для навчання у школі може послужити наступною сходинкою у підвищенні комп'ютерної грамотності вчителів та новою сторінкою у веденні уроків – наприклад, музики, яка завжди була більше орієнтована на слухові образи, ніж на візуальні. Проте, завдяки використанню інформаційних технологій з'явилася можливість зобразити звук, показати його наочно, як фізичне явище. Що не може не захоплювати.

Отже, для виконання вчителем поставленої задачі йому потребується

спеціалізований програмний засіб, на прикладі якого він зможе зобразити аудиторії, як виглядає звуковий сигнал та у чому взагалі полягає сутність проблеми сприйняття та аналізу звуку. Головним результатом розробки такого ПЗ є створення інтуїтивно зрозумілого та швидкого рішення, яке спростить навчальний процес як учнів, так і вчителя [2], та дозволить додавати уроки, тести та використовувати спеціальний модуль для обробки звуку, який можна задіяти у навчальному процесі.

Створений веб-додаток передбачує наявність ряду вимог:

- можливість реєстрування та логування;
- можливість перегляду елементів курсу, їх змінення;
- адміністративна панель для користувача групи ‘superuser’ (адміністратора);
- можливість введення даних та отримання результату у вигляді діаграм аудіоспектру.
- дизайн повинен бути адаптивним;
- один користувач повинен мати не більше ніж одну роль.

Мета дослідження – розробка вебдодатку для автоматизації процесу проведення занять з музики та перевірки знань учнів у середній школі.

Об’єктом дослідження є використання вебсервісів у навчальному процесі в школі.

Предметом дослідження є впровадження комп’ютерних технологій на уроках музики.

Для реалізації мети дослідження необхідно вирішити наступні задачі:

- а) аналіз предметної області. Надання характеристики навчальним комп’ютерним програмам, їх видам;
- б) дослідження існуючих рішень, аналіз їх особливостей, переваг та недоліків;
- в) обрання методології та написання аналітичної частини;

- г) проектування програмного компоненту, обрання стеку технологій;
- д) тестування готової програми.

Створений веб-додаток передбачує наявність ряду вимог:

- можливість реєстрації та авторизації;
- можливість перегляду елементів курсу, їх редагування;
- адміністративна панель для користувача групи ‘superuser’ (адміністратора);
- можливість аналізу аудіо файлів за допомогою діаграм аудіоспектру.
- дизайн повинен бути адаптивним;
- один користувач повинен мати не більше ніж одну роль.

У цій роботі використовувалися як теоретичні, так і емпіричні методи наукового дослідження: метод аналізу, синтезу, класифікації, спостереження, порівняння, узагальнення.

Виходячи з поточної ситуації у освітній сфері та її стрімкої спрямованості на перехід до хмарних технологій та вебсервісів у найближчому часі, можна сказати, що завдяки вдало обраній предметній області тема дослідження є актуальною.

1 ТЕОРЕТИЧНІ АСПЕКТИ КОМП'ЮТЕРНИХ НАВЧАЛЬНИХ ПРОГРАМ ТА СПОСОБІВ ЇХ ВИКОРИСТАННЯ

1.1 Поняття комп'ютерної навчальної програми

Навчальна комп'ютерна програма – це термін, який використовується для будь-якого комп'ютерного програмного забезпечення, створеного з освітньою метою. Він охоплює різні діапазони від програмного забезпечення для вивчення мови до програмного забезпечення для керування класом і програм-довідок. Мета всього цього програмного забезпечення – зробити освітню галузь більш швидкою та ефективною [3].

Є декілька основних типів реалізації комп'ютерних навчальних програм або платформ, і обидва з них пов'язані із веб-технологіями, тому що десктопні програми на сьогодні набагато менш актуальні, ніж раніше, так як уся індустрія прямує до децентралізованих систем, а програма, встановлена на одному комп'ютері – це завжди централізована система. Навіть якщо розглядати у якості прикладу десктопне програмне забезпечення із постійним виходом до Інтернету, все одно веб-додаток, який можна запустити з браузера, завжди буде більш зручним варіантом. Хоча деякі компанії воліють випускати свій продукт у обидвох варіантах, щоб більш продвинуті користувачі, яким потрібен розширений функціонал, могли звернутися до завантажуваного десктопного додатку, а інші користувалися більш простою версією ПЗ у браузері.

Комп'ютеризовані системи навчання називають електронними освітніми технологіями, електронним навчанням, навчальною платформою або системою управління навчанням. Основна відмінність полягає в тому, що VLE (virtual learning environment) і LMS (learning management system) є додатками, тоді як навчальна платформа має спільні риси з операційною системою, поверх якої можна запускати різні освітні веб-додатки [3].

Терміни «віртуальне навчальне середовище» та «навчальна платформа» зазвичай використовуються для опису ряду інтегрованих веб-додатків, які надають вчителям, учням, батькам та іншим особам, які беруть участь у освіті, інформацію, інструменти та ресурси для підтримки та покращення навчання та управління. Ці терміни загалом є синонімами «керованих навчальних середовищ» і «керованих віртуальних навчальних середовищ» (MVLE) [4].

Програми, які є частиною цих онлайн-сервісів, можуть включати веб-сторінки, електронну пошту, дошки оголошень і дискусійні форуми, текстові та відеоконференції, спільні щоденники, онлайн-соціальні зони, а також інструменти оцінки, керування та відстеження.

VLE, такі як Moodle, характеризуються конструктивістськими підходами, їх основним призначенням є місце для співпраці та поширення дискусій, а не розміщення об'єктів навчання, які можна відстежувати. Мартін Дугіамас, засновник Moodle, заснував розробку на конструктивістських теоріях, і цей факт досі зазначається на сторінках Moodle [5].

Інший основний освітній VLE, Blackboard, відмічає на своєму сайті, що він спрямований на те, щоб «студенти були інформовані, залучені та співпрацювали разом» [6], знову ж таки, повторюючи конструктивістські принципи та принципи співпраці. Однак Blackboard називає себе LMS, що підкреслює плутанину навколо термінів у галузі.

А основна різниця – у тому, що LMS – це частіше внутрішньо корпоративні розробки з обмеженим функціоналом, а VLE діють як самостійні одиниці і можуть обслуговувати не одну установу.

1.2 Види комп'ютерних навчальних програм

Існує багато різних видів комп'ютерних навчальних програм, але можна виділити п'ять основних:

- електронні курси;

- помічники для класів;
- програми для оцінювання показників успішності;
- електронні довідники;
- індивідуальні програми.

1.2.1 Електронні курси

Електронний курс – це термін, який поєднує поняття "курс" і "програмне забезпечення". Спочатку він використовувався для опису додаткового навчального матеріалу, призначеного як набори для вчителів або тренерів або як навчальні посібники для студентів, зазвичай упакованих для використання з комп'ютером. Значення та використання терміну розширилося і може стосуватися всього курсу та будь-якого додаткового матеріалу, якщо використовується для посилання на онлайн-класи або класи у «комп'ютерному форматі». Багато компаній використовують цей термін, щоб описати весь «пакет», що складається з одного «класу» або «курсу» разом із різними уроками, тестами та іншим необхідним матеріалом. Самі навчальні програми можуть бути в різних форматах: деякі доступні лише в Інтернеті, наприклад, на веб-сторінках, а інші можна завантажити у вигляді файлів PDF або інших типів документів. Багато форм освітніх технологій тепер охоплюються терміном «курсіві програми». Більшість провідних освітніх компаній замовляють або включають курси до своїх навчальних пакетів [7].

1.2.2 Помічники для класів

Деякі навчальні програми призначені для використання в шкільних класах. Зазвичай таке програмне забезпечення може проектуватися на велику дошку в передній частині класу та/або одночасно працювати в мережі настільних комп'ютерів у класі. Найбільш помітними є дошки SMART, які використовують SMART Notebook для взаємодії з дошкою, що дозволяє

використовувати ручки для цифрового малювання на дошці. Цей тип програмного забезпечення часто називають програмним забезпеченням для управління класом. Хоча вчителі часто обирають використовувати освітнє програмне забезпечення з інших категорій у своїх ІТ-пакетах (наприклад, довідкові роботи, програмне забезпечення для дітей), виросла ціла категорія освітнього програмного забезпечення, спеціально призначеного для допомоги в навчанні у класі. У цій категорії брендинг був менш сильним, ніж у тих категорій, що орієнтовані на домашніх користувачів. Назви програмного забезпечення часто дуже спеціалізовані та виробляються різними виробниками, включаючи багато відомих видавництв освітніх книг.

1.2.3 ПЗ для оцінювання показників успішності

У зв'язку із нанесенням шкоди довкіллю та потребою, щоб заклади стали «безпаперовими», все більше навчальних закладів шукають альтернативні способи оцінювання та тестування, які завжди традиційно споживають величезну кількість паперу. Програмне забезпечення для оцінювання відноситься до програмного забезпечення, основною метою якого є оцінювання та тестування студентів у віртуальному середовищі. Програмне забезпечення для оцінювання дозволяє студентам складати тести та іспити за допомогою комп'ютера, зазвичай під'єданого до мережі. Потім програмне забезпечення оцінює кожен тест тесту та виводить результати для кожного студента. Програмне забезпечення для оцінювання доступне в різних методах доставки, найпопулярнішими є програмне забезпечення для самостійного розміщення, програмне забезпечення онлайн та ручні системи голосування. Доступні також фірмове програмне забезпечення та системи програмного забезпечення з відкритим кодом. Хоча технічно відноситься до категорії Courseware (див. вище), лабораторія оцінювання навичок є прикладом комп'ютерного програмного забезпечення для оцінювання з методологією PPA-2 (Планувати, доводити, оцінювати) для створення та проведення комп'ютерного онлайн-іспиту. Moodle – приклад програмного

забезпечення з відкритим кодом із компонентом оцінювання, який набирає популярності. Інші популярні міжнародні системи оцінювання включають Google Classroom, Blackboard Learn та EvaluNet XT.

1.2.4 Електронні довідники

Багато видавців друкованих словників та енциклопедій з середини 1990-х років займаються виробництвом навчально-довідкового програмного забезпечення. На ринку еталонного програмного забезпечення до них приєдналися як стартап-компанії, так і визнані видавці програмного забезпечення, зокрема Microsoft.

Першими комерційними довідковими програмними продуктами були переформулювання існуючого вмісту у видання CD-ROM, часто доповнені новим мультимедійним вмістом, у тому числі стисненим відео та звуком. У новітніх продуктах використовувалися інтернет-технології, щоб доповнити продукти CD-ROM, а нещодавно повністю замінити їх. Вікіпедія та її допоміжні частини (наприклад, Вікісловник) знаменували новий відрив у освітньому довідковому програмному забезпеченні.

Раніше енциклопедії та словники складали свій зміст на основі запрошених та закритих команд спеціалістів. Концепція Wiki дозволила розробити спільні довідкові роботи через відкриту співпрацю, залучаючи як експертів, так і неекспертів.

1.2.5 Індивідуальні програми

Деякі виробники вважали звичайні персональні комп'ютери невідповідною платформою для навчання програмному забезпеченню для дітей молодшого віку і замість них виробляли індивідуальні пристрої, зручні для дітей. Апаратне та програмне забезпечення зазвичай об'єднуються в один продукт, наприклад, схожий на дитячий ноутбук. Клавіатура ноутбука для дітей молодшого віку дотримується алфавітного порядку і порядку QWERTY

для старших. Найбільш відомим прикладом є продукти Leapfrog. Сюди входять оригінально розроблені ручні консолі з різноманітними під'єднуваними навчальними ігровими картриджами та електронними пристроями, схожими на книжки, в які можна завантажувати різноманітні електронні книги. Ці продукти є меншими за розміром, ніж портативні комп'ютери, але мають набагато обмеженіший діапазон цілей, зосереджений на грамотності.

У той час як основні операційні системи розроблені для загального використання і більш-менш налаштовані для навчання лише доданими до них наборами програм, різноманітні виробники програмного забезпечення, особливо дистрибутиви Linux, прагнули надати інтегровані платформи для спеціальної освіти.

До цього виду програм також можна додати програмне забезпечення для корпоративних тренінгів, автошкол, вивчення мов, сервіси із елементами гейміфікації – тобто, усе те, що спеціалізовано суто під свої цілі.

1.3 Існуючі рішення, їх особливості, переваги та недоліки

Навчальне програмне забезпечення – це майже необхідна для здорового функціонування навчальної установи система, і будь-яка організація повинна мати можливість проводити онлайн-навчання. Існує багато програмних рішень, але трьома лідерами на ринку все одно до сих пір залишаються Moodle, Blackboard та Canvas. Кожен з цих сервісів має більш-менш спільний функціонал та ряд переваг та недоліків, про який зазначено нижче.

1.3.1 Особливості системи Moodle

Moodle – це відкрите програмне забезпечення для управління навчанням, яке базується на модульному дизайн [8]. Це дає змогу

адміністраторам та викладачам створювати власні курси за допомогою плагінів. Moodle надає надійний набір функціональних можливостей і спільне навчальне середовище, які допомагають полегшити як викладання, так і навчання. Завдяки настроюваним функціям керування Moodle використовується для створення власних веб-сайтів з онлайн-курсами для викладачів і тренерів для досягнення цілей навчання. Moodle дозволяє розширювати та адаптувати навчальні середовища за допомогою плагінів, створених спільнотою. Усі плагіни у Moodle написані на мові PHP.

Moodle пропонує спеціальний завантажуваний мобільний і настільний додатки. Використовуючи мобільний додаток, ви можете мати офлайн-доступ до свого контенту, сповіщень, календаря, нагадувань та швидкого переходу між контентом. Крім того, якщо куплений пакет Branded Moodle App, можна завантажити персоналізовану версію своєї програми для настільних комп'ютерів. Пакет доступний для операційних систем macOS, Windows і Linux.

Moodle надає універсальну навчальну платформу (див. рис. 1.1), дозволяючи налаштувати бажаний спосіб навчання викладача. Він може легко інтегруватися з низкою інших додатків і програм для задоволення конкретних потреб педагога.

Список функцій Moodle включає звіти та аналітику, персоналізовані інформаційні панелі, керування даними, контроль через панель адміністратора тощо. Це створює непоганий навчальний простір для студентів. Крім того, Moodle дозволяє ділитися файлами та завантажувати інтерактивні курси в систему.

Переваги:

- багатий функціонал завдяки великій кількості модулів, накопиченій протягом довгих років існування спільноти Moodle;
- підтримка різних мовних пакетів;
- список клієнтів включає Cisco, Bank of America та інші компанії, що свідчить про досить високу якість продукту;

- доступна підтримка: підтримка в реальному часі, телефон, електронна пошта та квитки;
- є гарний безкоштовний тарифний план для учнів та вчителів.

Укрїнська (uk) Вхід

Система електронного забезпечення навчання ЗНУ

Search Courses

НОВИНИ [Перевірте актуальність свого е-мейлу. Втрата доступу до системи Moodle - посилання](#)

Новини сайту

Втрата доступу до системи Moodle ЗНУ
Кириченко Наталя Володимирівна - Wednesday 28 October 2020 09:05 AM

Шановні користувачі СЕЗН ЗНУ (Moodle)!

Якщо ви забули пароль і вказали електронну адресу в профілі системи Moodle ЗНУ, то використовуйте посилання для відновлення паролю
<https://moodle.znu.edu.ua/mod/page/view.php?id=133015>
Перевіряйте папку СПАМ

Якщо ви забули логін для входу в систему СЕЗН ЗНУ, то пишть лист за адресами:

для студентів

- moodle.znu@znu.edu.ua

для викладачів:

- suprznu@gmail.com - Кириченко Наталя Володимирівна
- krivoyaroslav21@gmail.com - Кривий Ярослав Вячеславович

Зворотний зв'язок

- Оцінка якості курсу

Навігація

- На головну
- Новини сайту
- Статистика вибору дисциплін в межах освітньої прог...
- Статистика вибору дисциплін студентами на 2021-2022
- Вибір дисциплін на 2021-2022
- Рекомендації щодо використання технологій та засоб...
- Інструкція - основи роботи з системою Moodle
- Консультації для викладачів
- Нормативна база СЕЗН
- Корисні матеріали
- Корисні посилання
- Відновлення пароля
- Памятка для користувача системи Moodle
- Система оцінки курсу
- Оцінка якості курсу

Рисунок 1.1 – Зовнішній вигляд інтерфейсу Moodle

Недоліки:

- незважаючи на великий функціонал, у Moodle дуже перевантажений зайвими деталями інтерфейс, і людині без досвіду роботи з цим VLE важко буде розібратися у системі, тому це ставить під сумнів доцільність впровадження цієї системи до навчальних закладів, особливо якщо враховувати кількість викладачів старшого віку, які мають проблеми із користуванням складними комп'ютерними системами та учнів, не кожен з яких звик до систем управління навчанням;
- хоча різні мовні пакети і підтримуються, це зовсім не значить, що усі вони мають офіційну підтримку від виробника. Крім того, кожен із них необхідно дозавантажувати окремо. А деякі плагіни і досить не переведені на усі мови, що може спричинити певні незручності;

– великий послужний список компанії може казати як про якість продукту, так і про застарілість проектів, які не займаються активним розвитком та не приділяють достатню увагу побажанням клієнтів та втіленню нових ідей;

– ексклюзивні тарифні плани дуже дорогі для звичайних користувачів – так, вартість Branded Moodle App може доходити до 4000 доларів США [8], що, звісно, є невід’ємною сумою для будь-якого вчителя (див. рис. 1.2).

	Standard Moodle App			Branded Moodle App
	Free	Pro €199 per year	Premium €499 per year*	
Access course content and activities	✓	✓	✓	✓
In-App Messaging	✓	✓	✓	✓
Calendar	✓	✓	✓	✓
Gradebook	✓	✓	✓	✓
Multimedia Push notifications	✗	✓	✓	✓
Basic custom branding via CSS	✗	✗	✓	✓
Customisable app features	1 custom menu item 1 disabled feature 10 custom strings	2 custom menu items 2 disabled features 20 custom strings	Unlimited	Unlimited
Number of active devices for mobile push notifications	50	500	Unlimited	Unlimited
Offline access to content	2 courses / device	4 courses / device	Unlimited	Unlimited
Your own App in the stores	✗	✗	✗	✓
Separated Notifications Server	✗	✗	✗	✓
Full app branding (logo, colours)	✗	✗	✗	✓
Google Analytics	✗	✗	✗	✓
Mobile App Support via email	✗	✗	✗	✓

* Free for qualified sites (hosted by Moodle Partners, MoodleCloud and NGO), more info below

Рисунок 1.2 – Опис тарифних планів з офіційного сайту Moodle

1.3.2 Особливості системи Blackboard Learn

Blackboard Learn – це сучасний, інтуїтивно зрозумілий спосіб управління навчанням, який полегшує користування віртуальними платформами для навчання. Він забезпечує систему управління курсами, яка має відкриту архітектуру (див. рис. 1.3). Система може бути поєднана з інформаційною системою студента та процесами аутентифікації.

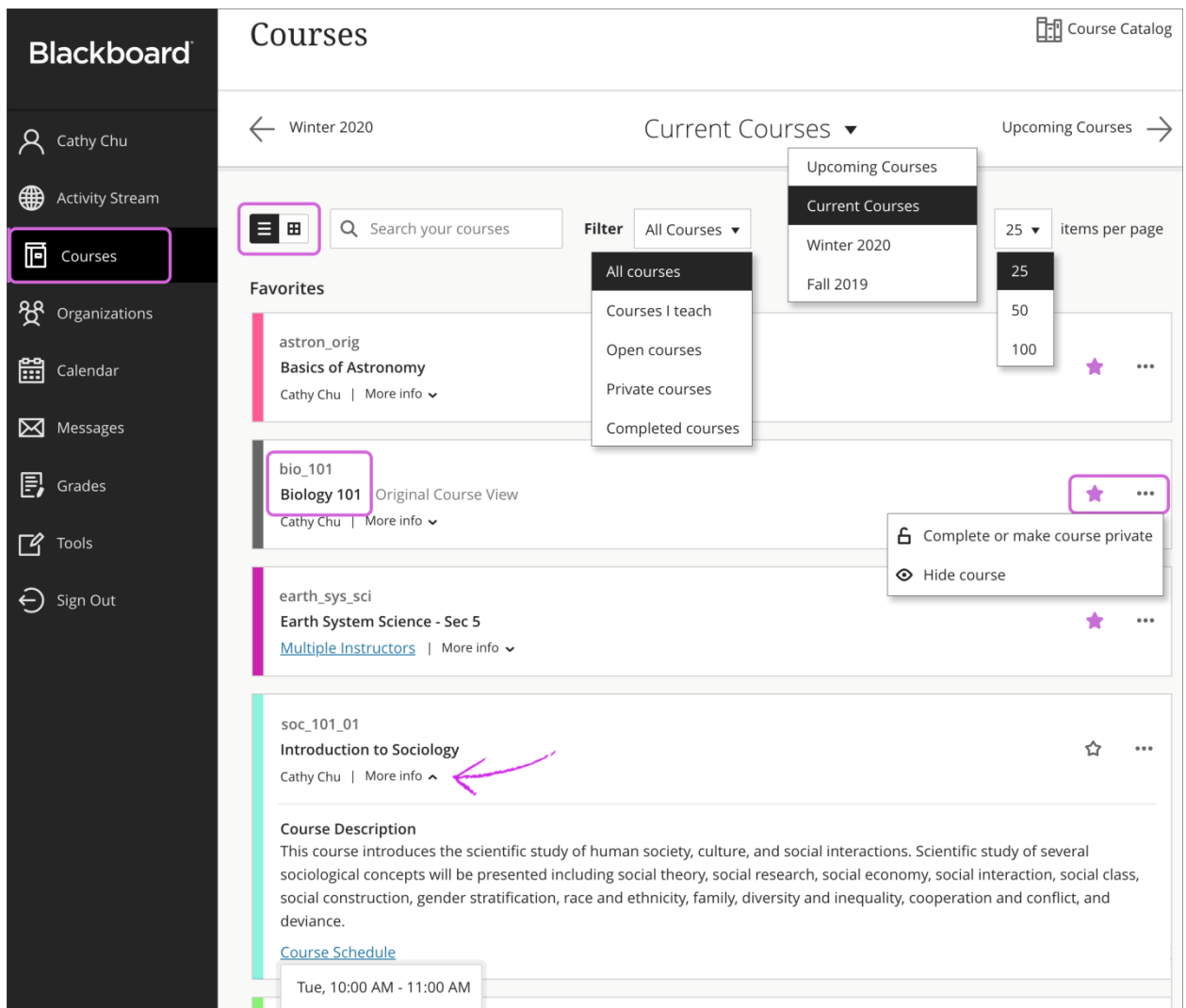


Рисунок 1.3 – Інтерфейс LMS Blackboard Learn

Blackboard пропонує повністю адаптивний дизайн для всіх пристроїв, включаючи мобільні телефони, планшети та планшети. Ця система пропонує спеціальні мобільні додатки для Android та iOS для підтримки таких видів

діяльності як спільна робота та оцінювання. Інтеграція важлива для розширення функціональності програми. Крім того, це має бути безперебійним процесом, щоб він не порушував уже поточні процеси. Blackboard можна легко інтегрувати з різними інструментами, включаючи BrainHoney, Epsilen, Pearson LearningStudio, Moodle Learning Management System, Sakai Collaborative and Learning Environment, itsLearning тощо.

Blackboard надає платформу, яку можуть використовувати навчальні заклади, бізнес-будинки та державні установи. Він пропонує повний набір функцій, таких як керування даними, попередній перегляд студентів, спільна інтеграція, SafeAssign, календарі та багато іншого.

Переваги:

- багатий та інтуїтивно зрозумілий інтерфейс, розроблений для того, щоб його могли використовувати навіть дошкільнята та учні початкових класів;
- перевіреність часом: дуже багато шкіл у США замовляють послуги цього сервісу, що також доказує його ефективність у веденні шкільного навчального процесу;
- підтримка тренінгів, квитків та номерів;
- відстежуваний прогрес завдань, добре налагоджений зв'язок учнів та вчителів.

Недоліки:

- підтримка лише англійської мови;
- все ж таки, спрощений функціонал у порівнянні із Moodle, хоча все необхідне є у наявності;
- закритий код, на відміну від Moodle;
- дуже великий ціник – Blackboard Learn не розголошує ціну на свої послуги, але якщо зв'язатися із підтримкою сайту напряму, то можна дізнатися: цей сервіс надає свої послуги за 9500 доларів США на рік, що робить його навіть дорожчим за Moodle, де є професійний тарифний план за 500 доларів. Хоча Blackboard Learn і надає тимчасовий доступ усім

бажаючим, але є абсолютно безкоштовна версія Moodle, яка поступається хіба що у плані інтерфейсу;

– погані оцінки мобільного додатку у Google Play, що свідчить про недостатню зацікавленість компанії відгуками користувачів щодо якості їх продукту.

1.3.3 Особливості системи Canvas

Canvas – це сервіс управління навчанням з відкритим кодом, який сьогодні є однією з найбільш швидкозростаючих систем. Він спеціально створений для шкільних закладів – для дітей будь-якого віку, починаючи від 5 років, – та вищих навчальних закладів. Ця платформа спрямована на те, щоб краще залучати користувачів до їхніх процесів викладання та навчання [9].

Canvas можна використовувати як у мобільних, так і у комп'ютерних браузерах. Він також має повний набір мобільних користувацьких програм для справжнього мобільного навчання та викладання. Крім того, він має функцію відеоконференцій під назвою Arc.

Canvas – ідеальна платформа для змішаного навчання. Він пропонує відкриту архітектуру, що призводить до безмежної інтеграції. Він може легко підключатися до комплексних освітніх інструментів, а також до існуючої інформаційної системи студентів. Фактично, зараз це призводить до інтеграції LTI між постачальниками LMS.

Canvas надає навчальну платформу для шкіл, вищих навчальних закладів і бізнес-будинків (див. рис. 1.4). Це полегшує змішане навчання завдяки своєму гнучкому та легкому для масштабування рішенням. Він підтримує надійне підключення та оновлення даних у режимі реального часу. Окрім цього, він підтримує запис або завантаження аудіо та відео, платформи для спільної роботи тощо.

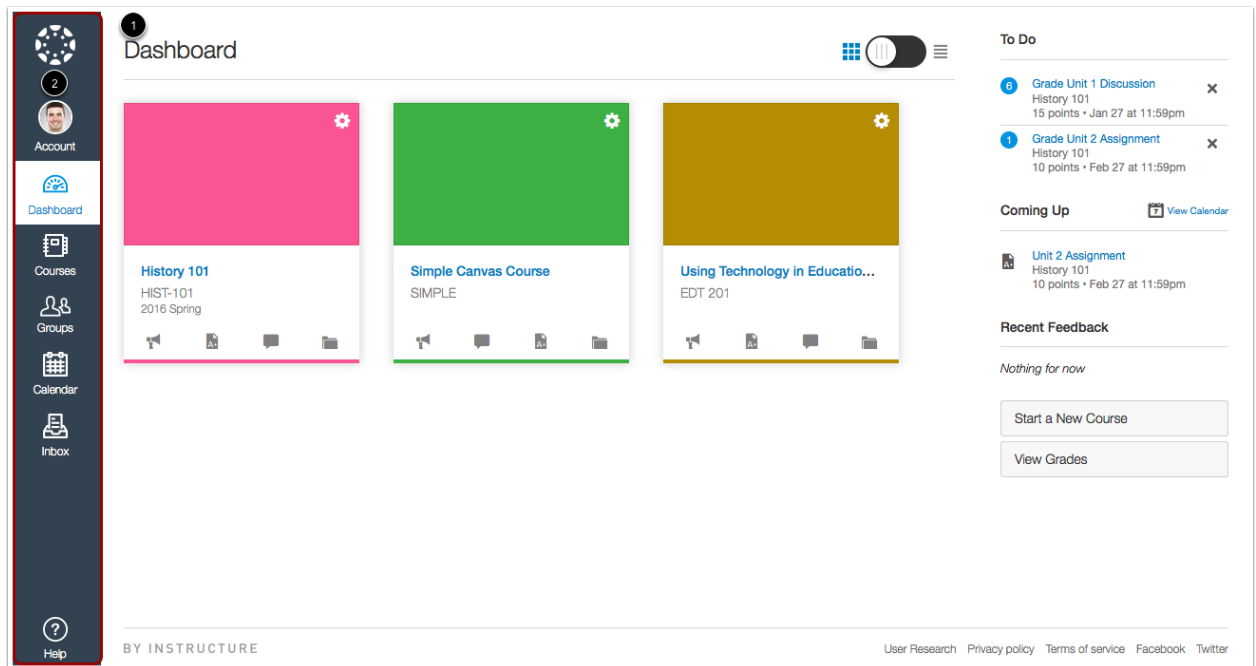


Рисунок 1.4 – Вигляд інтерфейсу Canvas

Переваги:

- Canvas має інтуїтивно зрозумілий інтерфейс, який робить увесь процес навчання цікавим та ефективним;
- сервіс багатомовний і підтримує такі мови як англійська, українська, китайська, німецька, російська, польська тощо;
- у цього сервісу є серйозні користувачі із високими рівнями акредитування – наприклад, університети штату Юта та Орегон, які служать певним маркером якості Canvas;
- завжди доступна онлайн-підтримка, телефон, електронна пошта, квитки та навчальні модулі, а також є повна підтримка хмарних технологій – таких як AWS, Microsoft та Google;
- безкоштовний тарифний план для вчителя, безкоштовний пробний період;
- завантажуваний додаток для мобільного телефону із високою оцінкою.

Недоліки:

- точна вартість підписки на сервіс невідома, але в залежності від кількості користувачів на курсі та об'єму інформації, що зберігається, вартість може складати від 120 до 1000 доларів на рік і більше. Це набагато менше, ніж за Blackboard Learn, але все ще дорого;

- інтерфейс не можна персоналізувати так, як у інших системах;
- відсутня функція навчання офлайн.

Хоча й існують інші навчальні системи – такі як, наприклад, Google Classroom та EvaluNet, але в них функціонал більш урізаний, і крім як для однієї чіткої задачі (задачі оцінювання учнів) ці сервіси навряд чи підходять. До речі, EvaluNet вже морально застарілий, і слід було б вибрати більш сучасне рішення у зв'язку з цим.

Отже, порівнюючи ці три сервіси, можна сказати, що найбільш якісними варіантами будуть Moodle або Canvas, тоді як Blackboard Learn потребує допрацювання та не вартий своїх грошей.

1.4 Висновки за розділом 1

Розглянувши ряд навчальних програмних засобів, можна дійти до висновку, що найкраща система управління навчанням (або віртуальне середовище навчання) – не та, яка коштує дорожче, а та, функціонал в якій та забезпеченість мовними пакетами дозволяє вести уроки без перешкод.

У розділі були розглянуті три найбільш популярні системи управління навчанням – це Moodle, Canvas від Instructure та Blackboard Learn. Багато де між собою ці веб-сервіси перекликаються, а авторські права на Blackboard і на Moodle належать майже одній компанії, але кожен із цих продуктів сповідує свою ідеологію. Якщо Moodle – це вже широко відоме народне рішення з відкритим вихідним кодом (за що його й люблять), то Blackboard – це більш бізнес-орієнтована варіація того ж самого середовища навчання, проте допрацьована у плані інтерфейсу. Canvas – взагалі інша історія.

Ідеальний кандидат для впровадження змішаної форми навчання, проте програє своїм конкурентам по функціоналу та вводить в деяку оману, не надаючи інформації про тарифні плани у відкритому доступі, як це робить Moodle.

Отже, якщо робити висновок, ні одне з існуючих на ринку рішень не задовольняє повністю потребам середньостатистичного викладача або студента по дистанційному навчанню, тому що усе упирається у гроші або складний інструментарій, з яким не кожен зможе ознайомитися. Саме тому найкращим рішенням буде взяти за основу спільні риси сервісів Moodle та Canvas і розробити свою персоналізовану навчальну систему, яка могла б задовольняти цілі ведення уроків музики у школі. Це дозволить зекономити кошти у перспективі та підштовхнути як учнів, так і вчителів не тільки до більш ефективного процесу навчання, а й до поглибленого освоєння комп'ютерних технологій.

2 АНАЛІЗ, ТЕХНОЛОГІЧНЕ ТА АЛГОРИТМІЧНЕ ЗАБЕЗПЕЧЕННЯ ПРОГРАМНОГО РІШЕННЯ

2.1 Сутність проблеми звукового аналізу

Звуковий аналіз (або аудіоаналіз) – це вилучення інформації та значення з аудіосигналів для аналізу, класифікації, зберігання, пошуку, синтезу тощо. Засоби спостереження та методи інтерпретації відрізняються, оскільки аудіоаналіз може стосуватися людського вуха та того, як люди інтерпретують чутний звук. Джерело, або це може означати використання такої технології, як аудіоаналізатор, для оцінки інших якостей джерела звуку, таких як амплітуда, спотворення, частотна характеристика тощо. Після того, як інформація джерела звуку була спостережена, розкрита інформація може бути оброблена для логічної, емоційної, описової чи іншої відповідної інтерпретації користувачем.

Найпоширенішою формою аудіоаналізу є слух людини. Тип сенсорного сприйняття, який зустрічається в більшій частині фауни планети, аудіоаналіз є фундаментальним процесом багатьох живих істот. Звуки, що видаються навколишнім середовищем або іншими живими істотами, забезпечують вхідні дані для слухового механізму, для чого мозок слухача може інтерпретувати звук і як він повинен реагувати. Приклади функцій включають мовлення, реакцію на страх, прослуховування музики тощо.

Слух, властивий людям, є основоположним у спілкуванні по всьому світу, а процес присвоєння значення й значення мовленню є складною, але необхідною функцією людського тіла. Вивчення слухової системи було зосереджено з використанням математики та аналізу синусоїдальних коливань і звуків. Перетворення Фур'є було важливою теоремою для розуміння того, як людське вухо обробляє повітря, що рухається, і перетворює його в діапазон чутних частот, приблизно від 20 до 20 000 Гц.[9]

Аудіосигнали можна аналізувати кількома різними способами, залежно від типу інформації, яку потрібно отримати від сигналу.

Типи аналізу сигналів включають:

- гучність і посилення;
- аналіз частотної області;
- частотна характеристика;
- коефіцієнт нелінійних спотворень та шум (THD+N);
- фаза;
- перехресні завади;
- інтермодуляційні спотворення, або інтермодуляція (IMD);
- стерео та об'ємний звук.

Апаратні аналізатори були основним засобом аналізу сигналів з моменту винаходу першого аудіоаналізатора, виробленого Hewlett-Packard, HP200A. Апаратні аналізатори, як правило, використовуються при розробці, тестуванні та виробництві професійних та споживчих продуктів. У міру розвитку комп'ютерних технологій інтегроване програмне забезпечення знайшло свій шлях до цих апаратних систем, а пізніше з'явилися інструменти аналізу звуку, які не потребували жодних апаратних компонентів, крім комп'ютера, на якому запущено програмне забезпечення. Програмні аналізатори звуку регулярно використовуються на різних етапах виробництва музики, таких як живий звук, мікшування та мастеринг. Ці продукти, як правило, використовують алгоритми швидкого перетворення Фур'є (FFT) та обробку, щоб забезпечити візуальне представлення аналізованого сигналу. Типи відображення та інформації включають частотний спектр, стереополе, поле об'ємного звучання, спектрограму тощо.

2.1.1 Аналізатор спектру та швидке перетворення Фур'є

Аналізатор спектру – це засіб для вимірювання спектрів, коливань, фізичних процесів. Аналізатор спектру вимірює величину вхідного сигналу в

залежності від частоти в межах повного діапазону частот приладу. Основне використання – вимірювання потужності спектру відомих і невідомих сигналів. Вхідний сигнал, який вимірюють більшість звичайних аналізаторів спектру, є електричним; однак спектральні композиції інших сигналів, таких як хвилі акустичного тиску та оптичні світлові хвилі, можна розглядати за допомогою відповідного перетворювача. Існують також аналізатори спектру для інших типів сигналів, наприклад, аналізатори оптичного спектру, які використовують для вимірювань прямі оптичні методи, такі як монохроматор.

Аналізуючи спектри електричних сигналів, можна спостерігати домінуючу частоту, потужність, спотворення, гармоніки, смугу пропускання та інші спектральні складові сигналу, які нелегко виявити у часовій області. Ці параметри корисні для характеристики електронних пристроїв, таких як бездротові передавачі. Дисплей аналізатора спектра має частоту по горизонтальній осі, а амплітуду - по вертикальній осі. Для випадкового спостерігача аналізатор спектру виглядає як осцилограф, і, насправді, деякі лабораторні прилади можуть функціонувати або як осцилограф, або як аналізатор спектру.

Існують два типи аналізаторів спектра: з розгорткою та ті, що використовують швидке перетворення Фур'є. Так як тема аудіоаналізу пов'язана із цифровим сигналом, то розглядатися буде другий варіант

З аналізатором спектру на основі ШПФ роздільна здатність по частоті дорівнює $\Delta \nu = \frac{1}{T}$, оберненому часу T , протягом якого вимірюється форма хвилі та перетворюється через перетворення Фур'є.

При аналізі перетворення Фур'є в цифровому аналізаторі спектру необхідно відбирати вхідний сигнал з частотою дискретизації ν_s , яка принаймні вдвічі перевищує пропускну здатність сигналу, через до межі Найквіста. Потім перетворення Фур'є створить спектр, що містить усі частоти від нуля до $\frac{\nu_s}{2}$. Це може поставити значні вимоги до необхідного

аналого-цифрового перетворювача та обчислювальної потужності для перетворення Фур'є, що робить аналізатори спектру на основі ШПФ обмеженими в діапазоні частот [10].

Аналізатор спектру реального часу не має сліпого часу – до деякого максимального діапазону, який часто називають «пропускною здатністю реального часу». Аналізатор здатний відбирати вхідний РЧ-спектр у часовій області та перетворювати інформацію в частотну область за допомогою процесу ШПФ. ШПФ обробляються паралельно, без проміжків і перекриваються, тому в обчисленому РЧ-спектрі немає пробілів і інформація не пропускається.

Аналіз спектру можна використовувати на звукових частотах для аналізу гармонік аудіосигналу. Типовим застосуванням є вимірювання спотворення номінально синусоподібного сигналу; синусоїда з дуже низьким рівнем спотворень використовується як вхідний сигнал для обладнання, що перевіряється, і аналізатор спектру може досліджувати вихід, який матиме додані продукти спотворення, і визначити відсоток спотворення для кожної гармоніки основної. Такі аналізатори свого часу називали «хвильовими аналізаторами». Аналіз можна проводити за допомогою цифрового комп'ютера загального призначення зі звуковою картою, обраною для відповідної продуктивності [11], і відповідним програмним забезпеченням. Замість використання синусоїди з низьким рівнем спотворень, вхідний сигнал можна відняти від вихідного сигналу, послабити та виправити по фазі, щоб отримати лише додаткові спотворення та шум, які можна проаналізувати [12].

Альтернативна методика, вимірювання повних гармонійних спотворень, компенсує основне за допомогою режекторного фільтра та вимірює загальний сигнал, що залишився, який є повним гармонічним спотворенням плюс шум; він не дає гармонійної деталізації аналізатора.

Аудіоінженери також використовують аналізатори спектру для оцінки своєї роботи. У цих програмах аналізатор спектру показуватиме рівні

гучності діапазонів частот у типовому діапазоні людського слуху, а не відобразатиме хвилю. У програмах живого звуку інженери можуть використовувати їх для визначення зворотного зв'язку.

2.1.2 Візуальне представлення спектру аудіохвиль

Спектрограма – це візуальне уявлення спектру частот сигналу, коли він змінюється з часом. При застосуванні до звукового сигналу спектрограми іноді називають сонографами, голосовими відбитками або голосограмами. Коли дані представлені в 3D-графіці, їх можна назвати водоспадами.

Спектрограма є основним інструментом спектрального аналізу звуку та інших галузей. Вона широко застосовується в аналізі мовлення. Спектрограму можна визначити як графік інтенсивності (зазвичай у логарифмічній шкалі – наприклад, у дБ) величини короткочасного перетворення Фур'є (STFT). STFT – це просто послідовність ШПФ віконних сегментів даних, де вікна зазвичай перекриваються в часі, як правило, на 25-50%. Це важливе представлення аудіоданих, оскільки людський слух заснований на своєрідній спектрограмі в реальному часі, що кодується вушним равликом. Спектрограма широко використовувалася в області комп'ютерної музики як керівництво під час розробки алгоритмів синтезу звуку. Під час роботи з відповідною моделлю синтезу узгодження спектрограми часто відповідає дуже хорошему узгодженню звуку. Фактично, синтез спектрального моделювання (SMS) заснований на синтезі короткочасного спектру безпосередньо за допомогою додаткових засобів [13].

Найбільш відомим форматом є графік із двома геометричними вимірами: одна вісь відображає час, а інша вісь – частоту; третій вимір, що вказує амплітуду певної частоти в певний момент часу, представлений інтенсивністю або кольором кожної точки на зображенні.

Існує багато варіацій формату: іноді вертикальна і горизонтальна осі перемикаються, тому час біжить вгору і вниз; іноді у вигляді діаграми водоспаду, де амплітуда представлена висотою тривимірної поверхні замість кольору чи інтенсивності. Осі частоти та амплітуди можуть бути лінійними або логарифмічними, залежно від того, для чого використовується графік. Звук зазвичай представляється з логарифмічною віссю амплітуди (ймовірно, в децибелах, дБ), а частота буде лінійною, щоб підкреслити гармонійні відносини, або логарифмічною, щоб підкреслити музичні, тональні відносини.

2.2 Методи наукового дослідження виконаної роботи

Наука являє собою класифіковані знання в межах будь-якого предмета. Процес, який допомагає систематичному та організованому одержанню інформації і створенню фрагментів знань на основі фактів у взаємозв'язку з іншими фактами, є описом і поясненням. Знання об'єктивні і піддаються перевірці; дійсність процедури може бути перевірена іншим дослідником.

Цілі будь-якої науки наступні:

- опис;
- пояснення;
- прогнозування;
- розуміння подій;
- контроль.

Керлінгер у 1972 році сказав, що наукова теорія визначається таким чином: «Теорія – це сукупність взаємопов'язаних конструкцій (термінів), визначень і тверджень, яка представляє систематичний погляд на явища через уточнення зв'язків між змінними з метою пояснити та передбачити такі явища» [14].

Наука як особлива форма знання має на меті розповісти й описати різноманітні структури й відносини у цьому світі й раціонально пояснити їх. Наукове знання відрізняється від повсякденного систематичним плануванням і систематичним упорядкуванням частин знань. Основу становлять повсякденні знання. Наукове знання перевершує його, намагаючись отримати загальні, об'єктивні та піддані перевірці знання, пов'язані з певним предметом. Основами наукового знання є теорія та емпіричні дослідження.

Дослідження можна охарактеризувати так:

- воно є процесом збору даних;
- воно систематичне;
- воно ставить під сумнів надані знання і робить синтез;
- воно веде до розширення знань;
- це поєднання всіх перерахованих вище характеристик.

Однією з вимог для проведення досліджень є також усвідомлення того, що дослідження повинні керуватися існуючими теоріями та гіпотезами, що впливають із теорій. Дослідження можна проводити за допомогою різних схем і моделей, які допомагають нам у застосуванні повторюваних і перевірених досліджень.

Наукова процедура означає процедуру, за допомогою якої послідовно реалізується задане завдання, пов'язане з дослідженням і досягнення мети дослідження. Наукова процедура базується на певній методології. Його зміст підкріплено методологією. Сенс наукової процедури полягає в успішному досягненні мети дослідження. Водночас наукова процедура означає впровадження методів дослідження.

Методи наукових досліджень можна поділити на 2 типи:

- теоретичні;
- емпіричні.

Теоретичні методи:

- використовують суто теоретичний підхід (аналіз, синтез, індукція, дедукція, моделювання);

- зазвичай не працюють з конкретними даними (числами);
- конкретні явища розглядаються з теоретичної точки зору.

Емпіричні методи:

- завжди працюють з конкретними даними;
- досягають конкретних частин знань за допомогою точних методів;
- їх предметами є живі предмети (вчителі, учні) або неживі предмети (підручники, реферати, написані учнями).

У цій роботі використовуються і теоретичні, і емпіричні методи дослідження, а саме метод аналізу, синтезу, класифікації, спостереження, порівняння, узагальнення.

Аналіз – шлях пізнання явища через пильне вивчення окремих його елементів.

Синтез – створення єдиної концепції з розрізнених даних, зібраних до купи.

Класифікація – розподіл зібраної інформації на тематичні блоки.

Спостереження – процес пасивного пізнання явищ, що вивчаються, і об'єктивної фіксації результатів сприйняття.

Порівняння – зіставлення предметів та явищ за значущими характеристиками.

Узагальнення – підведення до загального «знаменника» вивчених властивостей та приватних причинно-наслідкових рис предмета [15].

Для досягнення цілі кваліфікаційної роботи, а саме дослідження існуючих видів навчальних комп'ютерних програм були використані методи аналізу, класифікації, синтезу. При розробці програмного компонента були задіяні інші методи дослідження, а саме спостереження, порівняння та узагальнення, тому що під час написання та роботи з програмою був виконаний ряд замірювань та порівнянь із вже існуючими рішеннями. По деяких показниках результат перевищив очікування, тому можна звідси з'ясувати, що методи було обрано вірно.

2.3 Специфікація вимог програмного забезпечення

Специфікація вимог до програмного забезпечення (SRS) – це опис програмної системи, яка має бути розроблена. Змодельована вона відповідно до вимог бізнесу. Специфікація вимог до програмного забезпечення висвітлює функціональні та нефункціональні вимоги, і вона може включати набір варіантів використання, які описують взаємодії користувачів, які програмне забезпечення має надати користувачеві для ідеальної взаємодії [16].

2.3.1 Призначення

Призначення даної специфікації – побудувати навчальний програмний засіб для використання на уроках з музики та якісно покращити процес онлайн-навчання.

2.3.2 Аудиторія та потенційні користувачі продукту

Цей проект – програмний компонент, розроблений спеціально для процесу навчання учнів та вчителів (предметна область обрана за музичним спрямуванням), але може бути використаний і за межами навчальної установи у будь-якому колективі, який може мати на увазі роботу в команді.

2.3.3 Обсяг проекту

Мета усього проекту – побудувати безкоштовний, на відміну від конкурентів, зручний та інтуїтивно зрозумілий веб-сервіс, який допоможе вчителю обмінюватися матеріалами з учнями та дозволить задіяти модуль для обробки звуку у якості елемента гейміфікації на уроці.

Перш за все, програмний засіб повинен бути простим з точки зору дизайну, щоб принести максимально позитивний досвід користування сервісом. По-друге, система направлена на онлайн-освіту, тому вона повинна бути якомога простішою у виконанні, щоб людина із будь-якою якістю інтернет-зв'язку могла приєднатися та передивитися матеріали. По-третє, повинна бути можливість у будь-який момент за допомогою сучасних Docker-технологій розгорнути систему на допоміжному сервері або на шкільній P2P-мережі, якщо основний сервер не працює або на ньому відбуваються технічні роботи. Головна ціль – це забезпечення комфортного досвіду користувача усім, хто хоче вивчати музику онлайн у будь-який час.

2.3.4 Характеристики продукту

База даних розробленої системи зберігає наступну інформацію:

- дані користувачів (групу користувача, режими доступу до даних, логін, пароль, ім'я, прізвище, дату останньої авторизації, дату реєстрації, електронну поштову скриньку).
- права доступу різних груп користувачів до функціоналу сайту;
- Log адміністративної панелі;
- сесію користувача (разом із ключем та датою її закінчення);
- тип контенту та закріплену за ним модель;
- окрему таблицю для кожного навчального модуля, заповнену даними.

Для зображення побудованої системи у виді діаграм була використана мова моделювання UML [18].

Уніфікована мова моделювання (UML) – це мова моделювання загального призначення для розробки в області розробки програмного забезпечення, яка призначена для забезпечення стандартного способу візуалізації дизайну системи [19].

Модель сутність-відношення (або модель ER) описує взаємопов'язані речі, що представляють інтерес у конкретній області знань. Базова модель ER складається з типів об'єктів (які класифікують об'єкти, що представляють інтерес) і визначає відносини, які можуть існувати між сутностями (екземпляри цих типів об'єктів) [20].

У розробці програмного забезпечення модель ER зазвичай формується для представлення речей, які бізнес повинен пам'ятати для виконання бізнес-процесів. Отже, модель ER стає абстрактною моделлю даних, яка визначає структуру даних або інформації, яка може бути реалізована в базі даних, як правило, у реляційній базі даних.

ER-діаграма програми відображає логічну модель даних і побудована у наступному вигляді (див. рис. 2.1).

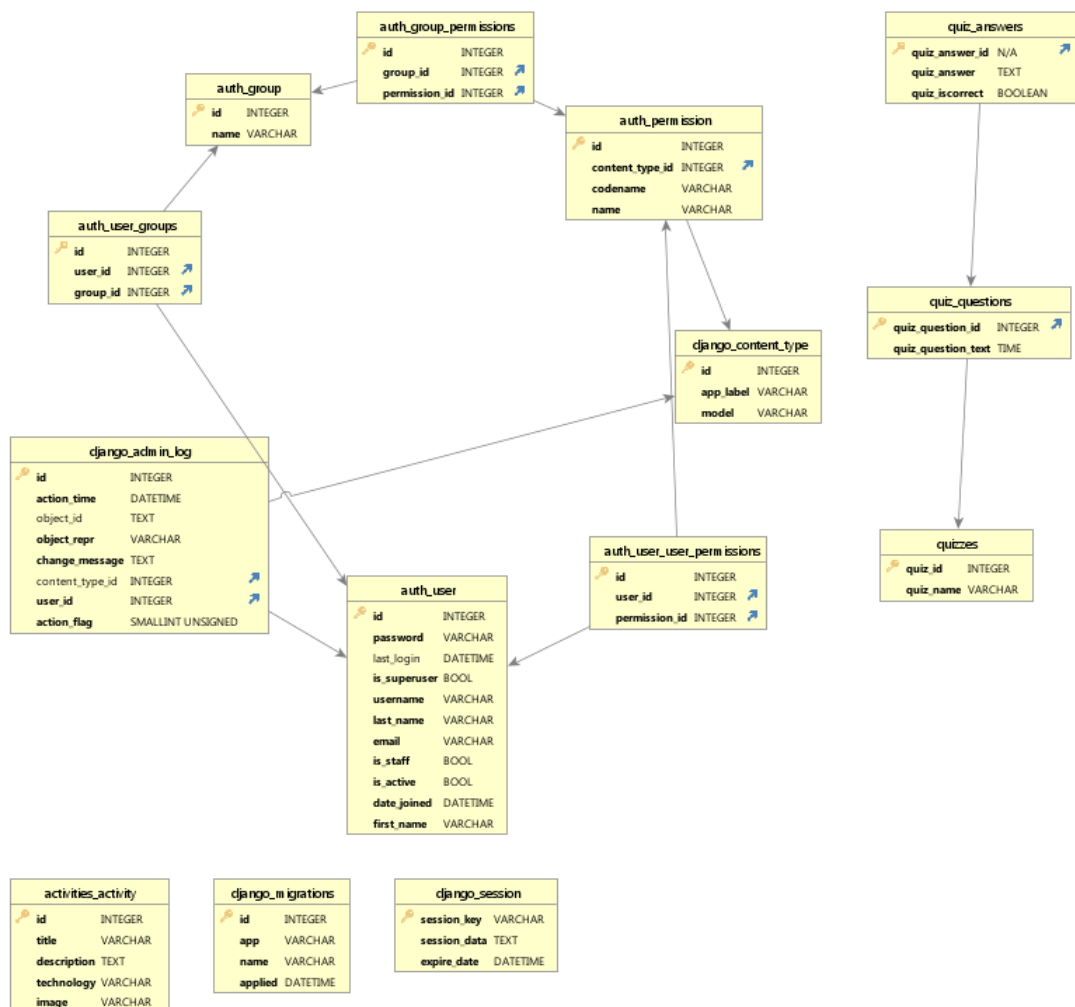


Рисунок 2.1 – ER-модель розробленої програми

Вона повністю відповідає створеній на SQLite базі даних (схемі), яка складається з 20 таблиць та може бути доповнена за необхідності. БД створювалася у безкоштовній системі адміністрування баз даних під назвою SQLite Manager, але пізніше була перенесена до іншого менеджера баз даних – DbVisualizer, тому що він більш наочний за зручний. Результати роботи із обома менеджерами баз даних зображені на наступних малюнках (див. рис. 2.2, 2.3, 2.4).

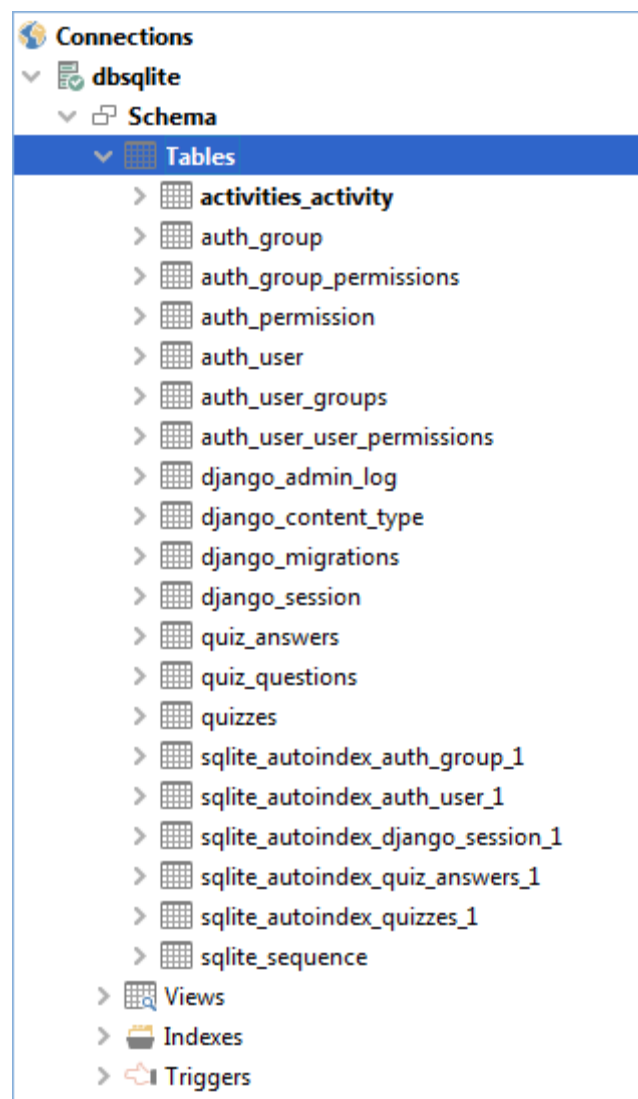


Рисунок 2.2 – Схема бази даних dbsqlite

TABLE_CAT	TABLE_SCHEM	TABLE_NAME	COLUMN_NAME	DATA_TYPE	TYPE_NAME	COLUMN_SIZE	BUFFER_LENGTH	DECIMAL_DIGITS	NUM_PREC_RADIX	NULLABLE
(null)	(null)	activities_activity	id	4	INTEGER	2000000000	2000000000	10	10	0
(null)	(null)	activities_activity	title	12	VARCHAR(100)	2000000000	2000000000	10	10	0
(null)	(null)	activities_activity	description	12	TEXT	2000000000	2000000000	10	10	0
(null)	(null)	activities_activity	technology	12	VARCHAR(20)	2000000000	2000000000	10	10	0
(null)	(null)	activities_activity	image	12	VARCHAR(100)	2000000000	2000000000	10	10	0

Рисунок 2.3 – Структура таблиці ‘activities_activity’

*	id	title	description
1	1	Урок 1	Тут ми навчимося створювати аналізатор аудіоспектрів на мові Python
2	2	Тест 1	Тест з теми "Аналіз аудіофайлів"
3	5	Модуль для роботи зі звуком	
4	6	Тест з музики про Моцарта	Lorem Ipsum is simply dummy text of the printing and typesetting industry. Lorem
5	8	VERYCHRISTMAS21312	41251251252162161261261

Рисунок 2.4 – Дані таблиці ‘activities_activity’

Діаграма розгортання – це тип діаграми UML, який показує архітектуру виконання системи, включаючи вузли, такі як апаратне або програмне середовище виконання, і проміжне програмне забезпечення, що з’єднує їх (див. рис. 2.5) [21].

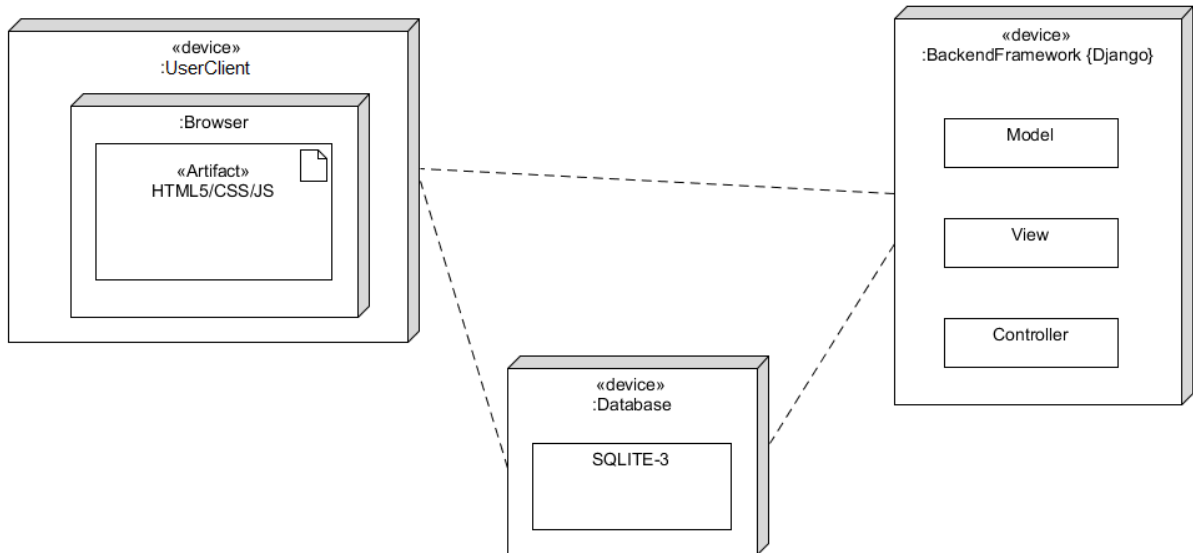


Рисунок 2.5 – Діаграма розгортання розробленої програми

2.3.5 Вимоги до розробки продукту

Створений веб-додаток передбачує наявність наступних функціональних вимог:

- можливість реєстрування та логування у додатку, а також змінення даних акаунту;
- можливість перегляду елементів курсу (уроків, тестів та інших видів активностей) усіма групами користувачів та їх додавання, змінювання або видалення адміністратором;
- адміністративна панель для користувача групи ‘superuser’ (адміністратора);
- можливість введення даних та отримання результату у вигляді діаграм аудіоспектру.

Також додаток передбачує наявність таких нефункціональних вимог:

- користувачі повинні переходити на будь-яку сторінку веб-додатку у три кліки (правило трьох кліків);
- дизайн повинен бути адаптивним для підтримки ПЗ на великих та

маленьких екранах;

- один користувач повинен мати не більше ніж одну роль;
- логування повинно відбуватися не більше ніж за 2 секунди.

2.3.6 Архітектура програмного засобу

Архітектура програмного забезпечення відноситься до фундаментальних структур програмної системи та дисципліни створення таких структур і систем. Кожна структура містить програмні елементи, відношення між ними та властивості як елементів, так і відносин. Архітектура програмної системи – це метафора, аналогічна архітектурі будівлі. Він функціонує як план системи та проекту розробки, викладаючи завдання, необхідні для виконання командами дизайнерів.

Зазвичай фреймворки використовують архітектуру Model-View-Controller (MVC). MVC популярний, оскільки він ізолює логіку програми від рівня інтерфейсу користувача, і різні частини програми можна розробляти окремо. Тут частина контролера – це програмний код, який керує взаємодією між моделлю та виглядом.

Фреймворк Django заснований на архітектурі Model-View-Template (MVT). MVT – це шаблон розробки програмного забезпечення для розробки веб-додатків. Це програмне забезпечення для веб-фреймворку діє як інструмент, який забезпечує спосіб створення та запуску веб-додатків. Django сам піклується про частину контролера, і нам не доведеться особливо турбуватися про те, як дані переміщуються між моделлю і представленням.

MVT складається з трьох компонентів – Model (модель), View (перегляд) і Template (шаблон).

Модель – це рівень абстракції для структурування та маніпулювання даними веб-додатка. Він діє як інтерфейс для підтримки даних. Це логічна структура даних, що лежить в основі всієї програми і допомагає обробляти базу даних.

Представлення – цей рівень інкапсулює логіку, відповідальну за обробку запиту користувача, і повертає відповідь. Це користувальницький інтерфейс для виконання логіки та взаємодії з моделями. Він відповідає за відображення всіх або частини даних для користувача. Представлення отримують дані, а також запит ("POST", "GET" тощо) від клієнта, а потім відповідно обробляють дані в базі даних через моделі. Ці дані доступні в інтерфейсі користувача за допомогою шаблонів

Шаблон – він забезпечує зручний для дизайнера синтаксис для відтворення інформації, яка буде представлена користувачеві. Шаблон містить статичні частини потрібного виводу HTML разом із деяким спеціальним синтаксисом, також відомим як Django Template Language (DTL), який описує, як буде представлено динамічний зміст веб-додатку.

2.3.7 Використані технології

HTTP – це протокол прикладного рівня в моделі набору протоколів Інтернет для розподілених, спільних гіпермедійних інформаційних систем. HTTP є основою передачі даних для всесвітньої павутини, де гіпертекстові документи містять гіперпосилання на інші ресурси, до яких користувач може легко отримати доступ, наприклад, клацанням миші або торканням екрана у веб-браузері.

Каскадні таблиці стилів (CSS) – це мова таблиць стилів, яка використовується для опису подання документа, написаного мовою розмітки. CSS є наріжною технологією всесвітньої мережі, поряд з HTML і JavaScript [24].

Python – це інтерпретована мова програмування загального рівня високого рівня. Його філософія дизайну підкреслює читабельність коду з використанням значних відступів. Його мовні конструкції, а також його об'єктно-орієнтований підхід спрямовані на те, щоб допомогти програмістам писати зрозумілий, логічний код для малих і великомасштабних проєктів.

Django – це безкоштовний веб-фреймворк на основі Python з відкритим вихідним кодом, який дотримується архітектурного шаблону модель–шаблон–перегляди (MTV). Він підтримується Django Software Foundation (DSF). Основна мета Django – полегшити створення складних веб-сайтів, керованих базою даних. Фреймворк підкреслює можливість повторного використання та «підключення» компонентів, менше коду, низьку зв'язність, швидкий розвиток і принцип «не повторюйся». Python використовується повсюдно, навіть для налаштувань, файлів і моделей даних. Django також надає додатковий адміністративний інтерфейс створення, читання, оновлення та видалення, який створюється динамічно шляхом інтроспекції та налаштовується через моделі адміністратора.

SQLite – це система керування реляційною базою даних (RDBMS), що міститься в бібліотеці C. На відміну від багатьох інших систем керування базами даних, SQLite вбудовується в кінцеву програму. Він є популярним вибором як програмне забезпечення для вбудованої бази даних для локального/клієнтського сховища в програмному забезпеченні, особливо у веб-сервісах.

Django nose – бібліотека, яка дозволяє швидко тестувати код програми за допомогою методу Test Discovery.

2.4 Висновки за розділом 2

Отже, у другому розділі розглянута аналітична частина роботи та технічне і алгоритмічне забезпечення розробленого програмного компоненту.

У якості алгоритму, який відповідає за побудову звукових хвиль аудіоспектра, служить швидке перетворення Фур'є. Роздільна здатність цього алгоритму рівнозначна формулі $\Delta \nu = \frac{1}{T}$. Звідси розраховується розмір звукових коливань, що поступають на вхід до аудіокарти.

Також завдяки компактній СУБД SQLite була створена модель даних, яка послужила основою для вбудованої в програмному засобі бази даних.

Окрім цього у розділі були розглянуті певні методи наукових досліджень (а саме, різні види теоретичних та емпіричних методів – це так звані аналіз, синтез, порівняння тощо), симбіоз яких дає максимально чітке представлення про те, у якому порядку треба розробляти програмну компоненту. Виходячи з цього, спочатку було виконано порівняння усіх існуючих рішень на ринку, а лише потім прийняте рішення про доцільність розробки і саму розробку власного навчального комп'ютерного програмного засобу для освіти у школах та освітніх закладах вищих рівнів акредитації.

3 РОЗРОБКА НАВЧАЛЬНОЇ ПРОГРАМИ ДЛЯ ВИКОРИСТАННЯ НА УРОКАХ МУЗИКИ

3.1 Загальний вигляд та функціонал розробленого веб-додатку

Робота адміністратору вебдодатку починається з форми авторизації (рис.3.2)

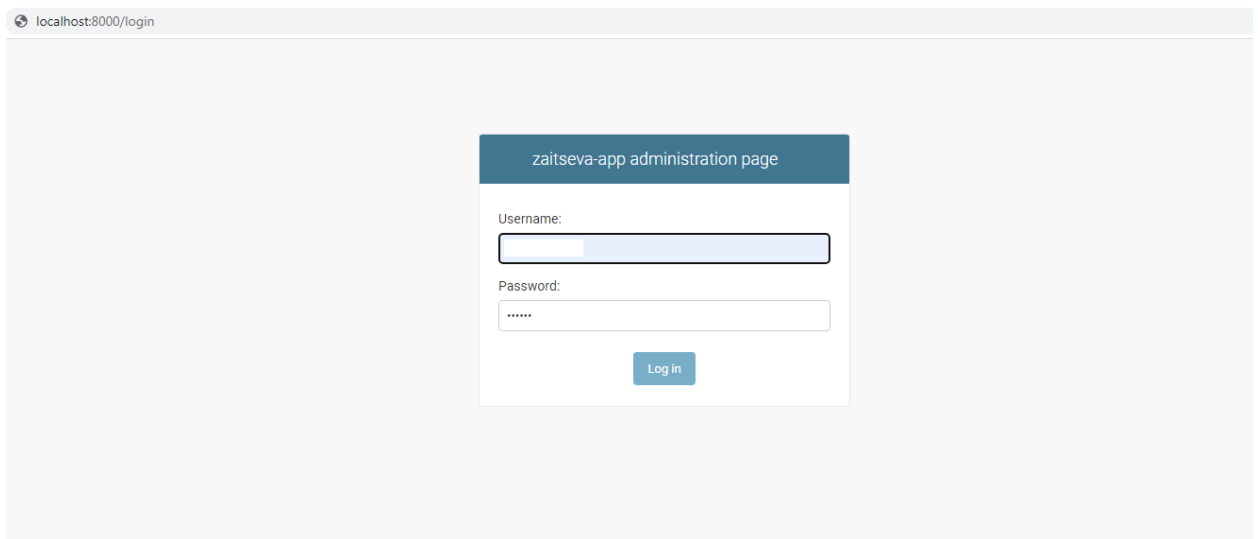


Рисунок 3.2 – Сторінка авторизації.

Далі користувача буде переадресовано до панелі адміністратора , з якої він може керувати розробленим програмним засобом та налаштовувати його на свій смак.

На наступному скриншоті можна побачити, що у панелі адміністратора відображаються усі останні зміни, які були зроблені у системі, колонка Recent actions (див. рис. 3.3).

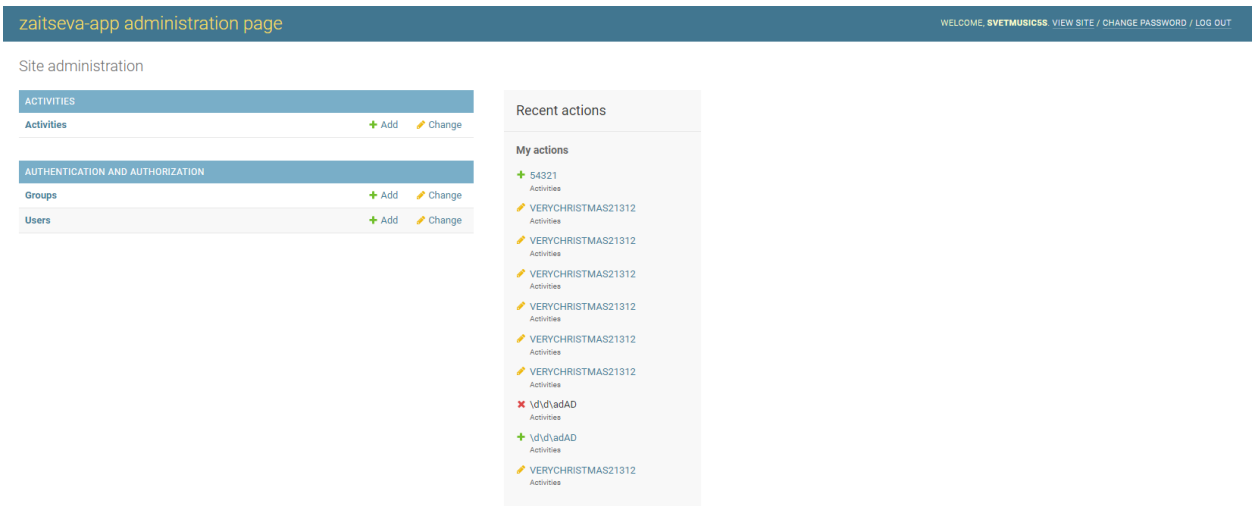


Рисунок 3.3 – Панель адміністратора розробленого додатку

У правому верхньому кутку можна побачити три кнопки:

- View site – перегляд сайту;
- Change password – змінити пароль;
- Log out – вийти з системи.

Ці елементи навігації дозволяють перейти до версії вебсайту очима звичайного користувача, змінити пароль користувача-адміністратора або вийти з аккаунту.

Пароль зберігається у БД SQLite, як і усі інші дані розробленого додатку, і якщо користувач захоче це зробити, то це краще робити напряму через панель Django, тому що він за умовчанням зашифрує пароль автоматично.

Django використовує PBKDF2 алгоритм з SHA256-хешем, механізм захисту паролів, рекомендований NIST. Цього повинно бути достатньо для багатьох користувачів: це безпечно та потребує великих об'ємів часу, якщо хтось захоче зламати аккаунт за допомогою брутфорсу [25].

Якщо перейти до розділу 'Activities' (див. рис. 3.4), то адміністратор може почати додавати або змінювати вже існуючі записи у базі даних системи. На цій сторінці можна додавати уроки та тести, а також модулі.

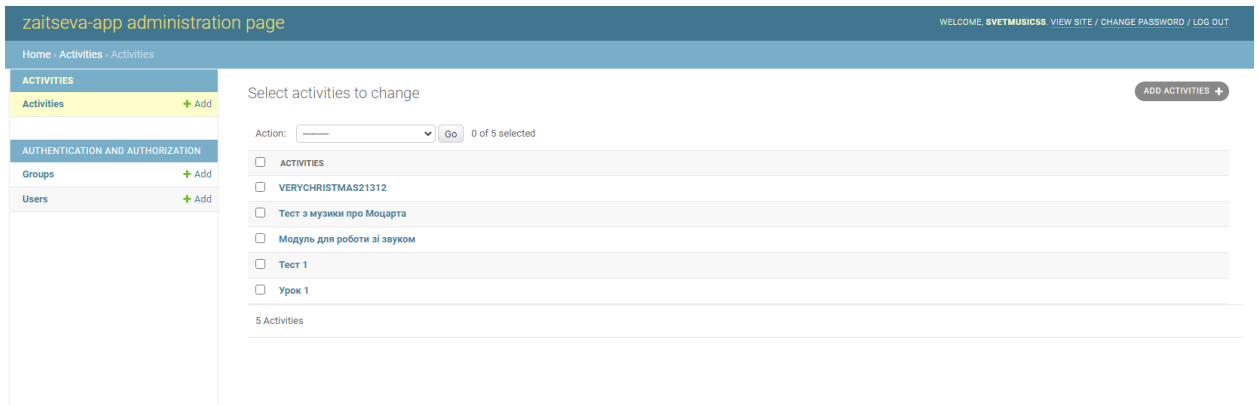


Рисунок 3.4 – Розділ ‘Activities’

При натисненні кнопки ‘Add activity’ адміністратора перенаправить до сторінки, де він зможе ввести дані (див. рис. 3.5), що він хоче додати до переліку існуючих записів.

Після заповнення сторінки необхідно натиснути кнопку ‘Save’, або ‘Save and add another’, якщо ви хочете додати ще один запис (див. рис. 3.6).

Рисунок 3.5 – Додання нової активності через меню ‘Add activity’

✔ The activities "Захист дипломної роботи" was added successfully.

Select activities to change

Action: 0 of 6 selected

<input type="checkbox"/>	ACTIVITIES
<input type="checkbox"/>	Захист дипломної роботи
<input type="checkbox"/>	VERYCHRISTMAS21312
<input type="checkbox"/>	Тест з музики про Моцарта
<input type="checkbox"/>	Модуль для роботи зі звуком
<input type="checkbox"/>	Тест 1
<input type="checkbox"/>	Урок 1

6 Activities

Рисунок 3.6 – Нову активність успішно додано

Перейдемо до сторінки 'Groups' та додамо нову групу користувачів (Students) так само, як раніше зробили з активністю (див. рис. 3.7, 3.8). Під час додавання груп можна надавати деякі привілегиї цілій групі користувачів – можна обрати, щоб користувачі могли логуватися до панелі адміністратора, але не могли нічого видаляти (тільки додавати), а можна зробити і навпаки – все залежить від побажань власника сервісу. На даний момент збережемо все як є і не будемо надавати ніяких привілегиї звичайним учням, щоб вони могли лише переглядати зміст сайту та не могли нічого змінювати.

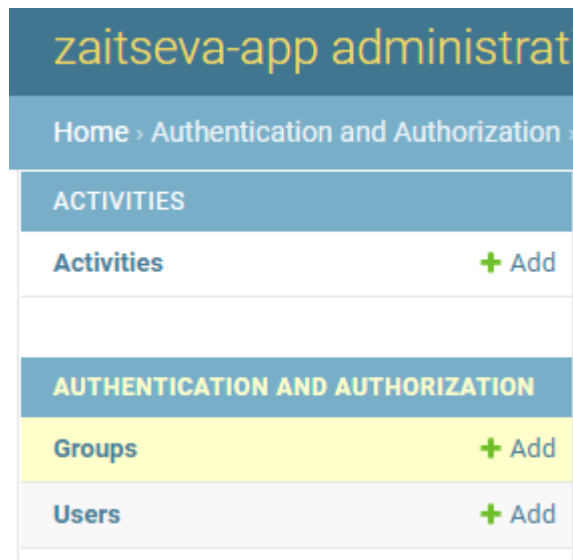


Рисунок 3.7 – Кнопка ‘Groups’ відображає існуючі групи користувачів

Add group

Name:

Permissions:

Available permissions ?

- activities | activity | Can add activity
- activities | activity | Can change activity
- activities | activity | Can delete activity
- admin | log entry | Can add log entry
- admin | log entry | Can change log entry
- admin | log entry | Can delete log entry
- admin | log entry | Can view log entry
- auth | group | Can add group
- auth | group | Can change group
- auth | group | Can delete group
- auth | permission | Can add permission
- auth | permission | Can change permission
- auth | permission | Can delete permission

Choose all ?

Hold down "Control", or "Command" on a Mac, to select more than one.

Chosen permissions ?

activities | activity | Can view activity

Remove all ?

Рисунок 3.8 – Додавання нової групи ‘Students’ із дозволом переглядати активності

Далі можна додавати користувачів до цієї групи (або вручну через базу даних, або через веб-додаток). Після цього перейдемо до меню ‘Users’ та спробуємо додати нового користувача (див. рис. 3.9).

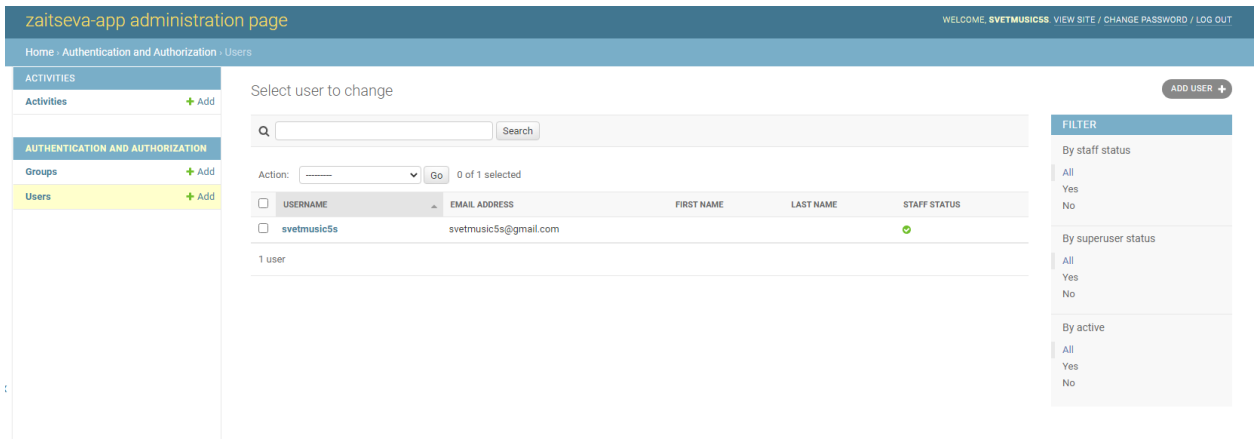


Рисунок 3.9 – Відображення усіх сторінок користувачів у системі

На цій же сторінці можна змінювати дані користувачів. У якості прикладу спробуємо змінити приватні дані адміністратора (див. рис. 3.10).

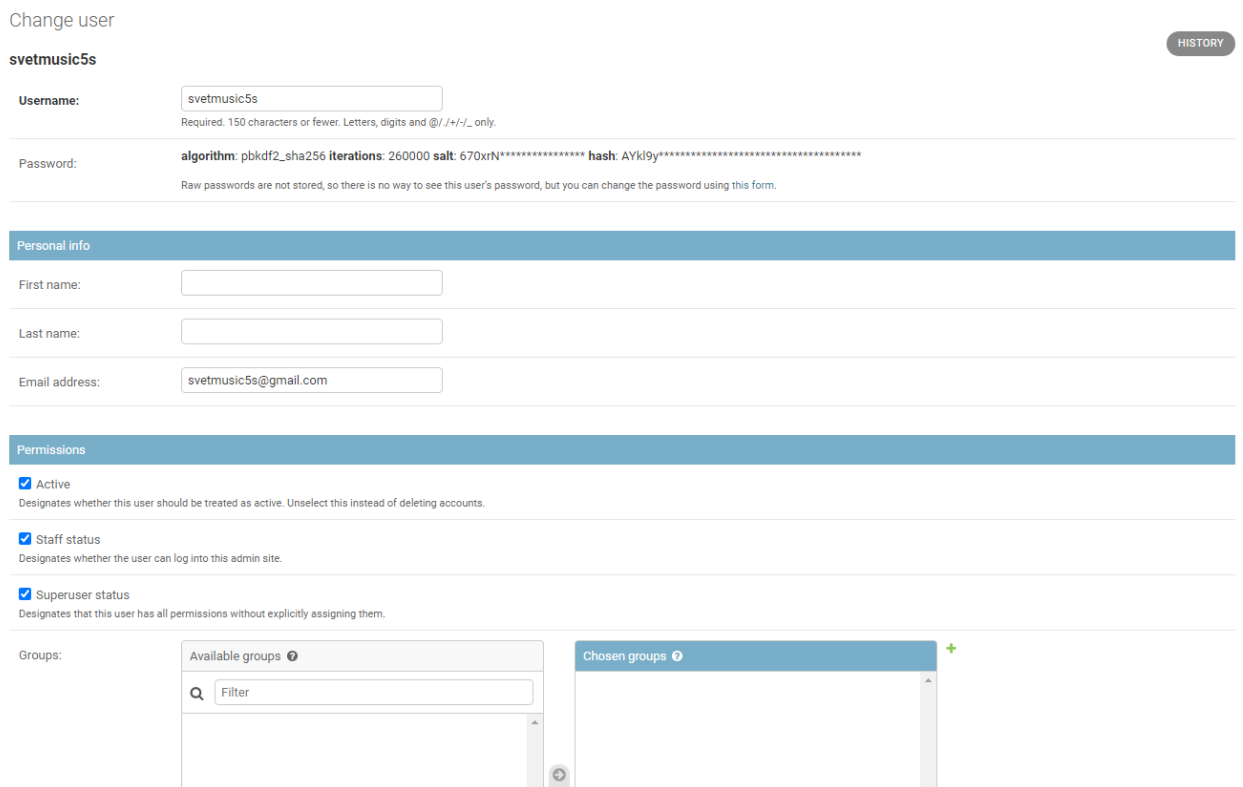


Рисунок 3.10 – Змінення даних користувача за допомогою кнопки ‘Change user’

Можна також повернутися до попереднього меню та спробувати додати нового користувача (див. рис. 3.11).

Add user

First, enter a username and password. Then, you'll be able to edit more user options.

Username:
 Required. 150 characters or fewer. Letters, digits and @/./+/-/_ only.

Password:
 Your password can't be too similar to your other personal information.
 Your password must contain at least 8 characters.
 Your password can't be a commonly used password.
 Your password can't be entirely numeric.

Password confirmation:
 Enter the same password as before, for verification.

Рисунок 3.11 – Додання нового користувача під іменем studentZNU

Система була розроблена з урахуванням того, щоб користувачі не створювали сторінки із надто популярними та простими паролями, тому якщо хтось спробує ввести у рядок пароль «111111», то у нього цього не вийде (див. рис. 3.12).

Add user

First, enter a username and password. Then, you'll be able to edit more user options.

Please correct the error below.

Username:
 Required. 150 characters or fewer. Letters, digits and @/./+/-/_ only.

Password:
 Your password can't be too similar to your other personal information.
 Your password must contain at least 8 characters.
 Your password can't be a commonly used password.
 Your password can't be entirely numeric.

This password is too common.
 This password is entirely numeric.

Password confirmation:
 Enter the same password as before, for verification.

Рисунок 3.12 – Додаток пропонує ввести інший пароль, щоб сторінка була захищена

Після успішного обрання пароля та логіна користувача буде додано (див. рис. 3.13).

Select user to change

Action: ----- 0 of 2 selected

<input type="checkbox"/>	USERNAME	EMAIL ADDRESS	FIRST NAME	LAST NAME	STAFF STATUS
<input type="checkbox"/>	studentZNU				✖
<input type="checkbox"/>	svetmusic5s	svetmusic5s@gmail.com			✔

2 users

Рисунок 3.13 – Оновлений список користувачів

Як можна побачити вище, після зміни пароля на більш оригінальний акаунт учня було успішно створено, і він не має прав доступу адміністратора. Це надає йому право передивлятися навчальні матеріали та користуватися усім, що є на самому сайті.

Після натискання на кнопку ‘View site’ або якщо просто перейти за стандартною адресою (localhost:8000), користувача перенаправить до самого навчального веб-сервісу (див. рис. 3.14).

Кожна активність має свій зміст, і він може редагуватися в залежності від потреби вчителя (див. рис. 3.15, 3.16). Уроки – це, по суті, лекції, тести перевіряють знання, отримані учнями с попередніх занять, а модуль для роботи зі звуком привносить певний елемент гейміфікації до цього веб-сервісу, щоб учні могли побачити, в яких галузях можна використовувати вивчені їми знання і як взагалі ком’ютери та інша техніка сприймає звуки (як і будь-яку функцію – у вигляді хвиль, коливань).

Елементи уроків

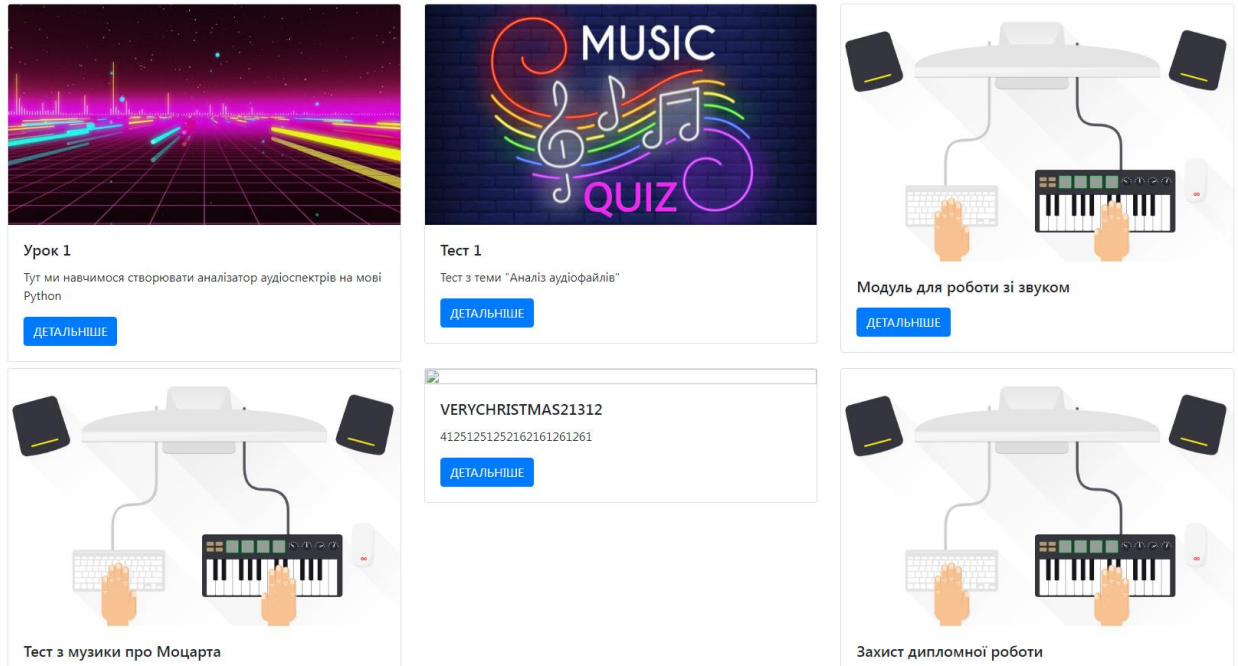


Рисунок 3.14 – Список усіх елементів уроків (активностей), які були створені

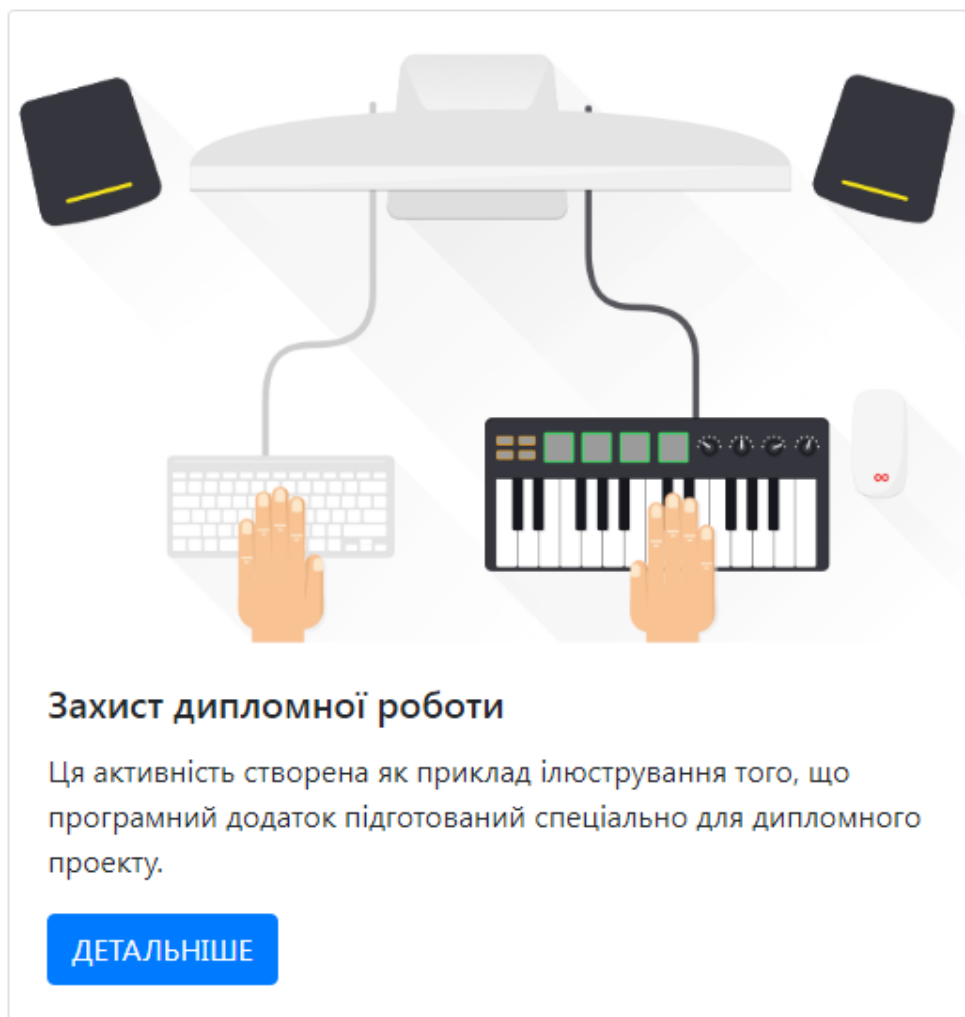


Рисунок 3.15 – Останній запис, який був доданий до бази даних

Урок 1



Опис:

Тут ми навчимося створювати аналізатор аудіоспектрів на мові Python

Технологія:

Python

Як створити аналізатор аудіоспектра?

```

Для початку нам потрібно отримати всі бібліотеки, які ми будемо використовувати:

from tkinter import messagebox
import pyaudio
import struct
import numpy as np
import matplotlib.pyplot as plt
import time
from tkinter import *

CHUNK = 1024 * 2
FORMAT = pyaudio.paInt16
CHANNELS = 1
RATE = 44100

p = pyaudio.PyAudio()

stream = p.open(
    format=FORMAT,
    channels=CHANNELS,
    rate=RATE,
    input=True,
    output=True,
    frames_per_buffer=CHUNK
)

fig, ax = plt.subplots()

x = np.arange(0, 2*CHUNK, 2)


```

Рисунок 3.16 – Урок 1. Як створити аналізатор аудіо спектру

Після виконання усіх інструкцій, отриманих у першому уроці, учні зможуть побачити згенерований на мові Python графік, який змінюється у реальному часі, в залежності від звуку, що поступає до мікрофону. В якості бібліотеки для аналізу звуку у цьому уроці була обрана бібліотека PyAudio –

вона найбільш проста на початкових етапах використання цієї мови та ознайомлювання із аналізом послідовностей звуку.

Після цього буде створений тест, за допомогою якого учні зможуть перевірити свої знання (див. рис. 3.17–3.19).



1 В якому місті, відомому з щорічного музичного фестивалю, народився Моцарт?

а. в Лейпцигу	в. у Відні
б. в Зальцбургу	г. у Мюнхені

2 Яку посаду займав при дворі архієпископа Зальцбурзького батько Моцарта?

а. камердинера	в. міністра
б. придворного музиканта	г. капелмейстера

3 Опера «Весілля Фігаро» написана на основі твору..?

а. Бомарше	г. Вольтер
б. <u>Піранес</u>	д. Гете
в.	

4 Останній твір Моцарта

а. Концерт для кларнету	в. Фуга органна
б. Меса Реквієм	г. Симфонія

Урок4

Тестування з теми " Творчість В.А. Моцарта"

ДЕТАЛЬНІШЕ

Рисунок 3.17 – Урок 4. Тест

Тестування з теми "Творчість В. А. Моцарта"

1/2 В якому місті, відомому з щорічного музичного фестивалю, народився Моцарт?

В Лейпцигу
 У Відні
 У Зальцбурзі

Назад
Далі

2/2 Опера "Весілля Фігаро" написана на основі твору:

Бомарше
 Вольтера
 Гете

Назад
Далі

Рисунок 3.18 – Проходження тесту з музики

Тест 2

2/2 Дякую за проходження тесту! Ви відповіли правильно!

Назад
Далі

Рисунок 3.19 – Тест виконано

Далі можна перевірити роботу модуля для аналізу звуку, розробленого за схожим до першого уроку принципу (проте, у цьому випадку код завдяки написанню додаткового класу компілюється у браузері).

Його особливість полягає у тому, що він аналізує аудіофайл, завантажений під час компіляції з коду, та відловлює різкі перепади гучності, і в моментах, коли гучність перевищує допустиму норму, записує звуковий сигнал у аудіофайл. На виході модуль дає можливість прослухати модифікований файл, позбавлений зайвих шумів, перепадів гучності тощо (див. рис. 3.20).

3.2 Тестування розробленої програми

Є декілька варіантів, як можна тестувати розроблене програмне забезпечення: мануально чи автоматично [27]. У випадку с Python Django існує гарний інструмент для автоматичного тестування, який називається `django nose`.

Щоб запустити його, достатньо завантажити пакет `django_nose` у проект за допомогою прописування у проекті `'django_nose'` у `INSTALLED_APPS` (див. рис. 3.21, 3.22).

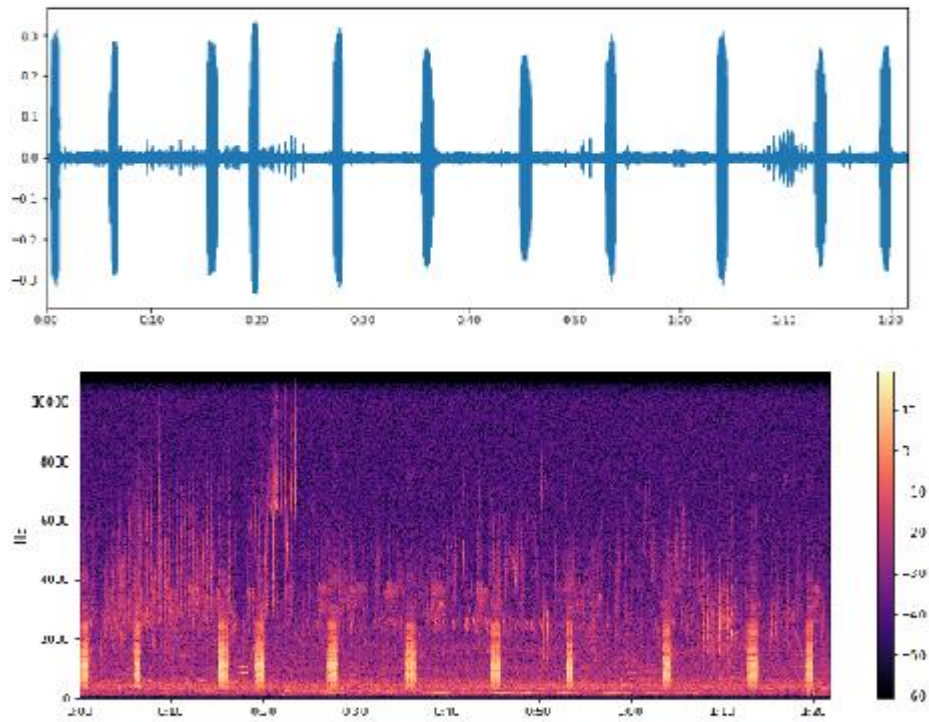
Модуль для роботи зі звуком



Опис:

Технологія:
Python

Як виглядає аналізатор аудіоспектра, скомпільований у декількох варіантах. Увага - файл читається з серверу!



Моменти різкого перевищення громкості відловлені бібліотекою librosa та збережені у файлі нижче:

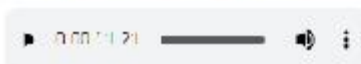


Рисунок 3.20 – Вигляд модулю для аналізу звуку

```

is_testing = 'test' in sys.argv
if is_testing:
    import coverage
    cov = coverage.coverage(source=['app'], omit=['*/tests/*'])
    cov.set_option('report:show_missing', True)
    cov.erase()
    cov.start()

execute_from_command_line(sys.argv)

if is_testing:
    cov.stop()
    cov.save()
    cov.report()

```

Рисунок 3.21 – Автоматичне тестування програми

```

python manage.py test
nosetests --cover-erase --cover-package=blog --cover-package=game --verbosity=1
Creating test database for alias 'default'...
.....
-----
Ran 19 tests in 3.586s

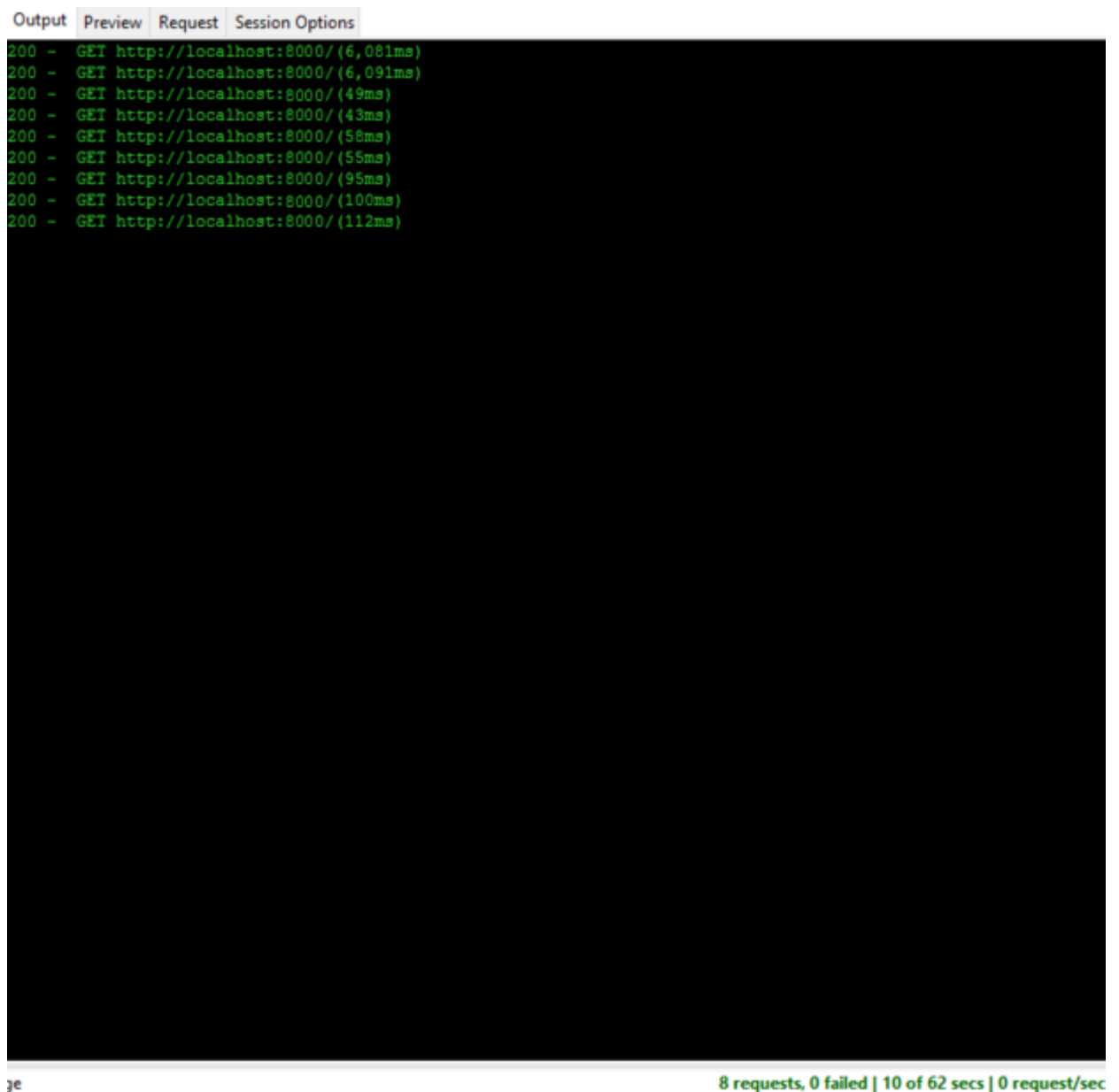
OK
Destroying test database for alias 'default'...
Name                               Stmts  Miss  Cover   Missing
-----
  /__init__.py                       0      0   100%
  /admin.py                          11      0   100%
  /apps.py                            3      3     0%   1-5
  /forms.py                           6      0   100%
  /migrations/0001_initial.py         9      0   100%
  /migrations/0002_postimage.py       7      0   100%
  /migrations/0003_auto_20170409_1343.py 5      0   100%
  /migrations/0004_auto_20170410_2116.py 5      0   100%
  /migrations/__init__.py             0      0   100%
  /models.py                          31      3    90%   9, 37, 45
  /tests.py                          127     0   100%
  /urls.py                             3      0   100%
  /views.py                           65      2    97%   15, 58

```

Рисунок 3.22 – Результат успішно пройденого тесту

Також можна перевірити можливість навантаження програми та швидкість її роботи за допомогою утиліти Web Surge. Щоб це зробити, достатньо завантажити пакет з сайту розробника.

На наступному скриншоті видно, що програмний компонент швидко реагує на GET-запити серверу, а тому витримує перевірку навантаження (див. рис. 3.23).



```
Output Preview Request Session Options
200 - GET http://localhost:8000/ (6,081ms)
200 - GET http://localhost:8000/ (6,091ms)
200 - GET http://localhost:8000/ (49ms)
200 - GET http://localhost:8000/ (43ms)
200 - GET http://localhost:8000/ (58ms)
200 - GET http://localhost:8000/ (55ms)
200 - GET http://localhost:8000/ (95ms)
200 - GET http://localhost:8000/ (100ms)
200 - GET http://localhost:8000/ (112ms)

8 requests, 0 failed | 10 of 62 secs | 0 request/sec
```

Рисунок 3.23 – Результат роботи Web Surge

3.2 Висновки за розділом 3

У цьому розділі було розроблено навчальну комп'ютерну програму для використання на уроках з музики.

Вона включає:

- можливість створення уроків, тестів та використання модулю для аналізу звуку;

- панель адміністратора із широким функціоналом, що дозволяє керувати користувачами, їх групами, матеріалами, завантаженими до розробленого додатку;
- можливість реєстрації та легування учнів і вчителів до веб-додатку.

ВИСНОВКИ

У даній кваліфікаційній роботі було розроблено програмний засіб для навчання музиці у школах. Програмний компонент включає у себе функціонал системи управління навчанням та модуль, що дозволяє аналізувати звуковий сигнал.

Для реалізації поставленої мети було вирішено ряд наступних завдань:

- аналіз предметної області. надання характеристики навчальним комп'ютерним програмам, їх видам;
- дослідження існуючих рішень, аналіз їх особливостей, переваг та недоліків;
- обрання методології наукового дослідження, аналітичний розділ;
- обрання стеку технологій, проектування програмного компоненту;
- тестування готової програми.

У результаті роботи над проектом була досягнута мета щодо підвищення ефективності навчання предмету музика у школі. Це підвищення досягається завдяки ряду переваг сервісу 'zaitseva-app' перед існуючими рішеннями на ринку – тобто, Moodle, Canvas, Blackboard Learn, Nearpod, Google Classroom.

Переваги включають:

- безкоштовну модель розповсюдження;
- простий та доступний інтерфейс;
- швидко, не перевантажену зайвим кодом систему.

Розроблений сервіс можна використовувати у будь-якому навчальному закладі, починаючи від шкіл і закінчуючи ВНЗ. В залежності від потреби цей ПЗ можна модифікувати, доповнювати різними новими видами активностей, функціоналом, що свідчить про його затребуваність на сьогодні. Ще одним фактором, що свідчить про його актуальність, є те, що на ринку немає навчальних сервісів для школи, які мають вбудовані інструменти для аналізу звуку [29].

ПЕРЕЛІК ПОСИЛАНЬ

- 1 Жарких Ю., Лисоченко С., Сусь Б., Третьак О. Комп'ютерні технології в освіті : навч. посібн. Київ : Видавничо-поліграфічний центр «Київський університет», 2012. 239 с.
- 2 Караванова Т. П. Методика використання навчальних комп'ютерних програм у навчальному процесі. URL : http://virtkafedra.ucoz.ua/el_gurnal/pages/vyp7/konf4/Karavanova.pdf (дата звернення : 12.10.21).
- 3 Пархуць Л., Ясинська С. Інформаційні системи в освіті: автоматизовані навчальні системи. *Гармонізація суспільства – новітній напрямок розвитку держави*: Матер. Всеукр. наук. конф. аспірантів та молодих вчених. Одеса, 2014.
- 4 Briefing paper 1: MLEs and VLEs explained. URL : https://www.webarchive.org.uk/wayback/archive/20080625160610mp_/http://jisc.ac.uk/uploaded_documents/bp1.pdf (дата звернення: 15.10.21).
- 5 About Moodle. URL : https://docs.moodle.org/311/en/About_Moodle (дата звернення: 15.10.21).
- 6 Blackboard Learn is for Educators. URL : <http://www.blackboard.com/learning-management-system/blackboard-learn.html> (дата звернення: 15.10.21).
- 7 Березовський В. С., Стеценко І. В., Завдський І. О. Створення електронних навчальних ресурсів та онлайнове навчання : навч. посіб. Київ: ВНУ, 2011. 208 с.
- 8 Introducing our new Moodle App plans. URL : <https://moodle.org/mod/forum/discuss.php?d=397926> (дата звернення: 16.10.21).
- 9 Building a Seamless Learning Ecosystem With Canvas & MasteryConnect. URL : <https://www.instructure.com/canvas/resources/case-study/new-caney-independent-school-district> (дата звернення: 16.10.21).

- 10 Oyvind Ryan Linear Algebra, Signal Processing, and Wavelets. Python Version – A Unified Approach – Springer Nature Switzerland AG 2019. 384 p.
- 11 Перекрест А., Чорний О., Гаврилець Г. Практикум з вивчення методів цифрової обробки сигналів у прикладних програмних пакетах: навч. посібник. Кременчук : ПП Щербатих О.В., 2015. 145 с.
- 12 ClariSonus Research Report #001, PC Sound Card Evaluation, John Atwood, 2006. URL : <https://web.archive.org/web/20110705033550/http://clarisonus.com/Research%20Reports/RR001-SoundCardEval/RR001-PCsoundCards.html> (дата звернення: 25.10.21).
- 13 Renardson audio designs: Distortion measurement. URL : <https://web.archive.org/web/20130625064334/http://www.angelfire.com/ab3/mjrap/golopid6.html> (дата звернення: 25.10.21).
- 14 Smith, J.O. Mathematics of the Discrete Fourier Transform (DFT) with Audio Applications, Second Edition, <http://ccrma.stanford.edu/~jos/mdft/>, online book, 2007 edition. URL : <https://ccrma.stanford.edu/~jos/mdft/Spectrograms.html> (дата звернення: 25.10.21).
- 15 Kerlinger F. N. Foundations of Behavioral Research. Praha: Academia, 2nd Edition. URL : <https://ia801603.us.archive.org/28/items/in.ernet.dli.2015.111808/2015.111808.Foundations-Of-Behavioral-Research-Edition-Second.pdf> (дата звернення: 29.10.21).
- 16 SEBERA, Martin. Selected Chapters on Methodology. 1st. Brno: Masaryk University, 2012. 52 p.
- 17 ISO/IEC/IEEE 29148:2018 – Systems and software engineering – Life cycle processes – Requirements engineering. URL : <https://www.sis.se/api/document/preview/80008109/> (дата звернення: 30.10.21).
- 18 ISO/IEC12207: 2017 «System and software engineering - Software life cycle processes». URL : <https://www.iso.org/obp/ui/#iso:std:iso-iec-ieee:12207:ed-1:v1:en> (дата звернення: 30.10.21).

- 19 ISO/IEC 19501:2005 - Information technology - Open Distributed Processing – Unified Modeling Language (UML) Version 1.4.2. URL : <https://www.omg.org/spec/UML/ISO/19501/PDF> (дата звернення: 30.10.21).
- 20 Blaha, Michael & Rumbaugh, James. Object-Oriented Modeling and Design with UML (2nd Edition). URL : <https://www.vtuplanet.com/d.php?type=notes&file=7th+Sem%2FObject+Oriented+Modeling+%26+Design+%28SJBIT%29+%5B10CS71%5D-NOTES.pdf> (дата звернення: 30.10.21).
- 21 Carte, Traci A.; Jasperson, Jon (Sean); and Cornelius, Mark E. Integrating ERD and UML Concepts When Teaching Data Modeling. Journal of Information Systems Education. Vol. 17 : Iss. 1, 2020. Article 9.
- 22 Deployment Diagram Tutorial. URL : <https://www.lucidchart.com/pages/uml-deployment-diagram> (дата звернення: 01.11.21).
- 23 Conolly T., Begg C. Database Systems: A Practical Approach to Design, Implementation, and Management. London: Pearson, 2015. 1400 p.
- 24 Chekalov A. P. Databases: from architecture to development. St.P.: Zarya, 2013. 384 p.
- 25 Official CSS Site. URL : <https://www.w3schools.com/css/> (дата звернення: 01.11.2021).
- 26 Password management in Django. URL : <https://docs.djangoproject.com/en/4.0/topics/auth/passwords/> (дата звернення: 01.11.2021).
- 27 Nickoloff J. Docker in Action. London: Portfolio, 2019. 566 p.
- 28 ISO 9000:2015 Quality management systems – Fundamentals and vocabulary (Системи управління якістю. Основні положення та словник термінів). URL : <https://khoda.gov.ua/image/catalog/files/%209000.pdf> (дата звернення: 01.11.21).
- 29 Зайцева С. Б., Решевська К. С. Цифрова обробка аудіоданих. Актуальні проблеми математики та інформатики : збірка тез доповідей Дванадцятої Всеукраїнської, дев'ятнадцятої регіональної наукової конференції молодих дослідників. Запоріжжя : Видавничий дім «Гельветика», 2021. С. 24-26.