



Міністерство освіти і науки України
Національний технічний університет України “Київський політехнічний
інститут імені Ігоря Сікорського”
Факультет інформатики та обчислювальної техніки
Кафедра інформаційних систем та технологій

Лабораторна робота №2
З дисципліни «Технології розроблення програмного забезпечення»
Тема: **«ДІАГРАМА ВАРІАНТІВ ВИКОРИСТАННЯ. СЦЕНАРІЙ
ВАРІАНТІВ ВИКОРИСТАННЯ. ДІАГРАМИ UML. ДІАГРАМИ КЛАСІВ.
КОНЦЕПТУАЛЬНА МОДЕЛЬ СИСТЕМИ»**
Музичний програвач

Виконав:
Студент групи ІА-23
Волошин В.В.

Перевірив:
Мягкий М. Ю.

Київ-2024

Зміст

Тема.....	3
Мета	3
Хід роботи	3
1. Схема прецедентів	3
2. Оберемо 3 прецеденти і напишемо для них сценарії використання	4
3. Схема класів.....	5
4. Структура бази даних.....	6
Висновки:	6

Тема:

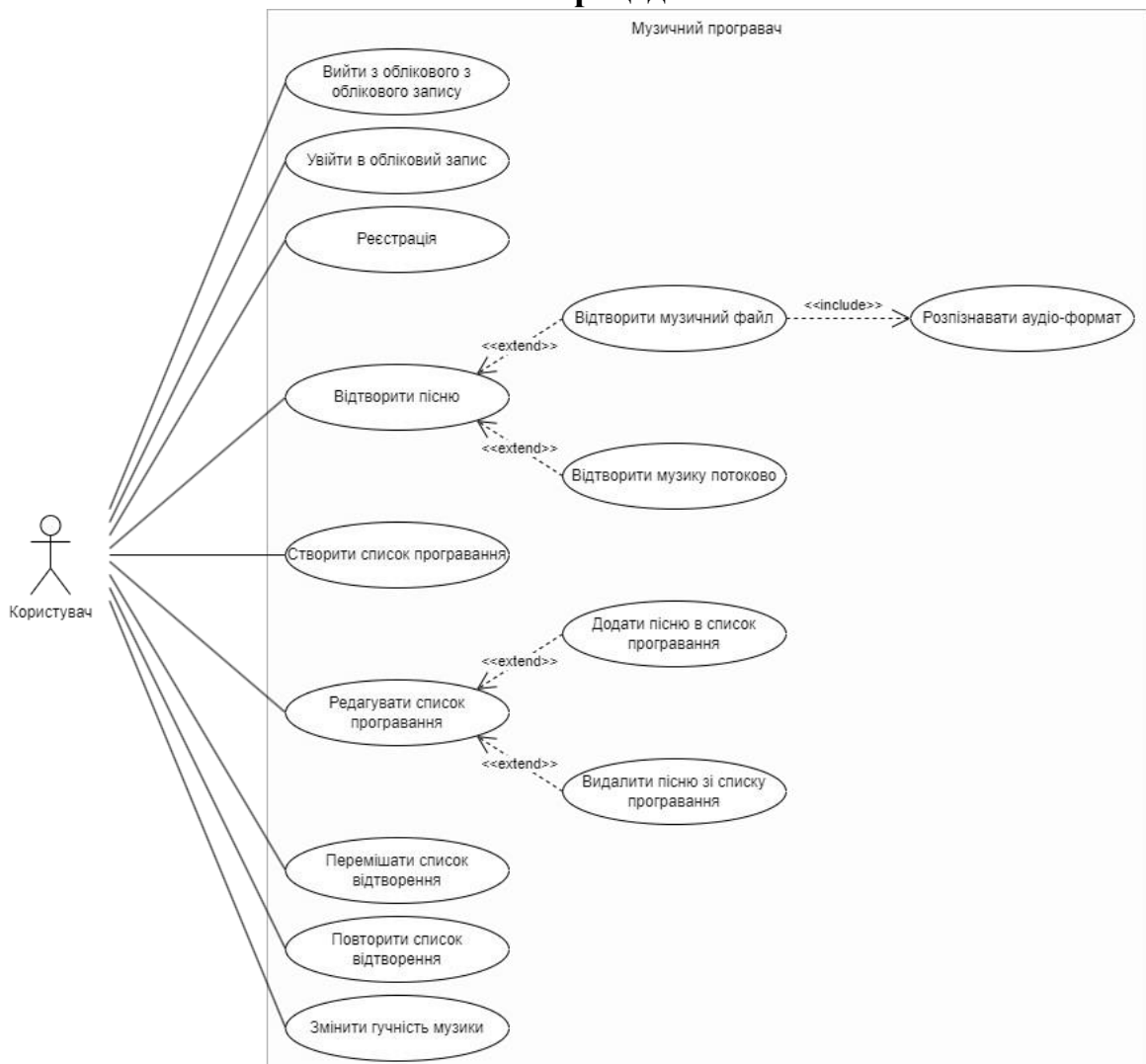
Діаграма варіантів використання. Сценарії варіантів використання. Діаграми uml. Діаграми класів. Концептуальна модель системи

Мета:

Метою даної лабораторної роботи є дослідження та практичне застосування UML-діаграм для моделювання варіантів використання системи та її концептуальної структури. Зокрема, робота зосереджується на побудові діаграм прецедентів, діаграм класів, а також створенні сценаріїв варіантів використання для системи. Вивчення цих моделей сприяє глибшому розумінню об'єктно-орієнтованого проектування та взаємодії між компонентами програмної системи.

Хід роботи

1. Схема прецедентів



2. Оберемо 3 прецеденти і напишемо для них сценарії використання

Сценарій використання 1: Відтворення пісні

Передумови: Користувач увійшов в обліковий запис.

Післяумови: Пісня буде відтворена на пристрої користувача.

Актори: Користувач.

Опис: Цей сценарій описує процес відтворення пісні в музичному програвачі.

Основний хід подій:

1. Користувач вибирає пісню зі списку або через пошук.
2. Система відкриває вибрану пісню і перевіряє її формат.
3. Якщо формат підтримується, система розпочинає відтворення пісні.
4. Користувач може зупинити або поставити пісню на паузу.

Винятки: Якщо пісня не підтримується (формат файлу невірний), система видає повідомлення про помилку.

Примітки: Відсутні.

Сценарій використання 2: Створення списку програвання

Передумови: Користувач увійшов в обліковий запис.

Післяумови: Новий список програвання буде збережений.

Актори: Користувач.

Опис: Цей сценарій описує процес створення нового списку програвання.

Основний хід подій:

1. Користувач вибирає опцію "Створити список програвання".
2. Система пропонує користувачу ввести назву для нового списку.
3. Користувач вводить назву та підтверджує.
4. Система зберігає новий список.

Винятки: Якщо користувач не введе назву, система видає повідомлення про помилку і пропонує ввести назву.

Примітки: Відсутні.

Сценарій використання 3: Додавання пісні в список програвання

Передумови: Користувач увійшов в обліковий запис. Користувач створив або обрав список програвання.

Післяумови: Пісня буде додана до вибраного списку програвання.

Актори: Користувач.

Опис: Цей сценарій описує процес додавання пісні в існуючий список програвання.

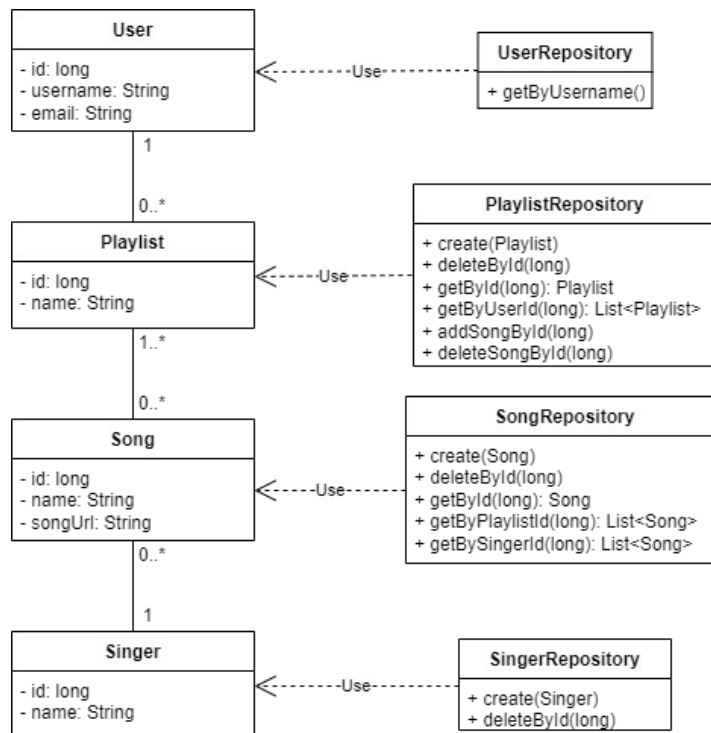
Основний хід подій:

1. Користувач вибирає пісню для додавання.
2. Система запитує, в який список програвання додати пісню.
3. Користувач вибирає потрібний список програвання.
4. Користувач підтверджує вибір.
5. Система додає пісню до вибраного списку.

Винятки: Якщо користувач не вибере список, система видасть повідомлення про помилку і запропонує вибрати список.

Примітки: Відсутні.

3. Схема класів



Ця діаграма класів представляє взаємодію між користувачем (**User**), плейлистом (**Playlist**), піснею (**Song**) та співаком (**Singer**), а також відповідними

репозиторіями для кожного з них. Кожен клас має свій репозиторій (наприклад, UserRepository, PlaylistRepository), який керує операціями зберігання й отримання даних. Взаємодія між класами показує, що користувач може створювати або використовувати плейлисти, які складаються з пісень, пов'язаних із певними співаками.

На даний момент у системі розроблено абстрактне відображення моделі даних, що може бути представлене відповідними інтерфейсами та класами з даної схеми.

4. Структура бази даних



Для реалізації цієї моделі даних використовується реляційна база даних із зв'язками "один до багатьох" між користувачами та плейлистами, піснями та виконавцями, а також зв'язком "багато до багатьох" між плейлистами та піснями через проміжну таблицю PlaylistSong.

Висновки:

У результаті виконання лабораторної роботи було успішно створено діаграму прецедентів для обраної системи, побудовано діаграму класів та концептуальну модель системи. Це дозволило краще зрозуміти структуру та взаємодію між різними елементами системи, а також забезпечило можливість реалізації ключових класів для роботи з базою даних за допомогою шаблону Репозиторію. UML-діаграми допомагають візуалізувати систему та полегшують процес її проектування та подальшої розробки.