

УДК 004.4

МОДЕЛЮВАННЯ МУЗИЧНОГО ПЛЕЄРА З ВИКОРИСТАННЯМ UML-ДІАГРАМИ КЛАСІВ

А.П. Волівач, кандидат технічних наук, доцент
Київський національний університет технологій та дизайну

В.В. Скідан, кандидат технічних наук, доцент
Київський національний університет технологій та дизайну

О.В. Учень, студент

Київський національний університет технологій та дизайну

М.В. Каменська, студент

Київський національний університет технологій та дизайну

Ключові слова: музичний плеєр, UML-діаграми класів, класи та методи, інтерфейс користувача, структура музичних файлів.

У сучасному світі популярним додатком серед молоді є музичний плеєр, який забезпечує користувачів інтуїтивно зрозумілими функціями відтворення музики, створення плейлистів, управління звуковими налаштуваннями та швидким доступом до треків.

Однак оптимальна організація відтворення плеєром особливих функцій (підключення інтерфейсу користувача, використання функціональних можливостей інтерфейсу користувача, процес відтворення даних тощо) потребує детального проєктування. Одним з таких рішень є застосування універсальної мови моделювання – UML (Unified Modeling Language), що широко використовується в об'єктно-орієнтованому програмуванні та є частиною уніфікованого процесу розроблення програмного забезпечення [1].

Для реалізації поставленої задачі було розроблено UML-діаграму класів музичного плеєра (рис. 1), яка відображає основні атрибути системи та операцій, необхідні для забезпечення функціональності.

Серед функцій музичного плеєра передбачено функції додавання та відтворення музичних файлів, відображення інформації про виконавців, навігацію між файлами, можливість перемотки музики, запуск та паузу треків.

Основні класи системи включають:

Controller – відповідає за управління основною логікою програми та наступні функціональні можливості:

- ініціалізацію програми та інтерфейсу користувача;
- керування музичним плеєром з використанням методів `Init()`, `InitUI()`, `LannelPlayer()`, `StopPlayer()`;
- оброблення натискання кнопок інтерфейсу користувача, відповідно для додавання, редагування, скасування та збереження змін: `(NewButton_Click(), EditButton_Click(), CancelButton_Click(), SaveButton_Click())`;
- відстеження стану програми через атрибути `IsInit`, `IsInitAI`,

IsPlayerStopped, IsPlayerStarted.

FileManager – забезпечує роботу з файлами:

– відкриває файли, дає їх представлення, зберігає та завантажує перевірку даних через методи OpenFile(), IsFileExists(), SaveDataToFile(), LoadDatafromfile());

– відстежує їх стан за допомогою атрибутів IsFileOpen, IsFileSaved, IsFileExist, FileData.

MusicArtist – містить інформацію про виконавців:

– атрибути Birthday, FullName, MusicalGenre, Name, Songs зберігають дату народження, повне ім'я, музичний жанр, ім'я та список пісень;

– метод AddSong() додає нові треки до списку.

MusicSong – містить інформацію про пісні, таку, як назву, рік випуску, альбом, виконавця, шлях до обкладинки та MP3-файл (атрибути Album, Artist, ArtistFullName, ArtworkPath, Mp3Path, Title, YearSong).

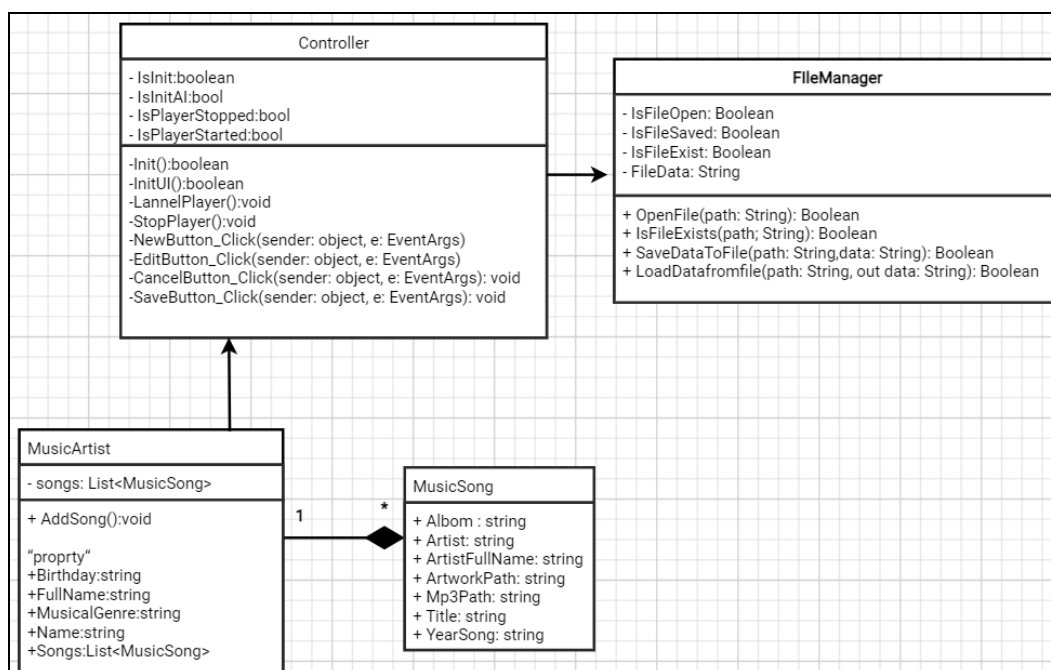


Рисунок 1 - UML-діаграма проектування музичного плеєра

Отже, розроблені класи UML-діаграми забезпечують ефективно та структуроване представлення основних компонентів системи, що включає логіку роботи музичного плеєра та управління музичними файлами. Такий підхід сприяє візуалізації зв'язків між класами, їхніми атрибутами та методами, що значно полегшує розуміння, проектування та підтримку систем подальшої розробки програмного забезпечення.

Список використаних джерел

1. Щербаков О. В. Основи об'єктно-орієнтованого програмування [Електронний ресурс] : навчальний посібник / О. В. Щербаков, Ю. Е. Парфьонов, В. М. Федорченко. – Харків : ХНЕУ ім. С. Кузнеця, 2019. – 237 с.