

МЕТОДИ ВІДТВОРЕННЯ ТА МОДЕРАЦІЇ МУЛЬТИМЕДІЙНОГО КОНТЕНТУ В DISCORD КАНАЛІ

Вінницький національний технічний університет

Анотація

У роботі представлено розробку бота для управління Discord каналом. Наведено UML-діаграму діяльності Discord бота. Запропоновано метод динамічного пошуку та керування відтворенням аудіоконтенту, а також метод модерації контенту.

Ключові слова: Discord канал, мультимедіа, відтворення аудіо, бот, модерація, сервер.

Abstract

The work presents the development of a bot for managing a Discord channel. A UML activity diagram of the Discord bot is provided. A method for dynamic search and playback control of audio content is proposed, as well as a method for content moderation.

Keywords: Discord channel, multimedia, audio playback, bot, moderation, server.

Вступ

В час технологічного прогресу виникає потреба у швидкому обміні інформацією. Однією з провідних платформ є Discord, яка об'єднує мільйони людей навколо себе. Великі Discord канали потребують постійного моніторингу та адміністрування для стабільної роботи серверів [1].

Боти в Discord часто мають обмежену функціональність, або не відповідають специфічним потребам конкретних спільнот [2]. Деякі з них мають складний процес налаштування або не дозволяють керувати медіа контентом, таким як музика, відео та голосові повідомлення. У цьому контексті розробка нового мультимедійного бота з розширеними можливостями для підтримки індивідуальних потреб різних спільнот є актуальною задачею.

Саме тому, тема має високу практичну значимість та відповідає сучасним тенденціям розвитку комунікаційних платформ. Її реалізація дозволяє підвищити ефективність управління Discord каналом та забезпечити зручність взаємодії користувачів.

Розробка методу відтворення мультимедійного контенту в Discord каналі

При створенні програмного забезпечення для мультимедійного бота, призначеного для управління Discord каналом, моделювання взаємодії користувача з ботом є одним з основних аспектів [3]. З цією метою розроблено UML-діаграму діяльності, яка демонструє основні сценарії використання бота, що показані на рисунку 1.

На діаграмі відображено повний цикл взаємодії користувача з Discord ботом. Починаючи від запуску бота на сервері та очікування на команди. Наступним кроком перевіряються права користувача на доступ до певного функціоналу. Доступ надається при певних умовах, після чого користувач може виконати дії: ввести назву пісні для відтворення. Також користувач може звернутись до окремих модулів бота, а саме: анонімний чат або чату репортів.

Після введення музичної команди бот виконує розпізнавання пісні. Виконується перевірка, чи користувач перебуває в голосовому каналі. Якщо так, то запускається відтворення мультимедійного контенту. Якщо користувач залишає голосовий канал, відтворення припиняється автоматично.

Для реалізації поставленої задачі було розроблено новий метод динамічного пошуку та керування відтворенням аудіоконтенту, який поєднує API-запити до зовнішніх мультимедійних платформ із фільтрацією результатів за ключовими словами та релевантністю. Він також реалізує адаптивну логіку взаємодії з користувачами в голосових каналах Discord. Метод враховує контекст попередніх запитів для

точнішого пошуку й інтерпретації команд, що підвищує зручність використання бота, зменшує кількість помилок і забезпечує швидке та релевантне відтворення медіаконтенту.

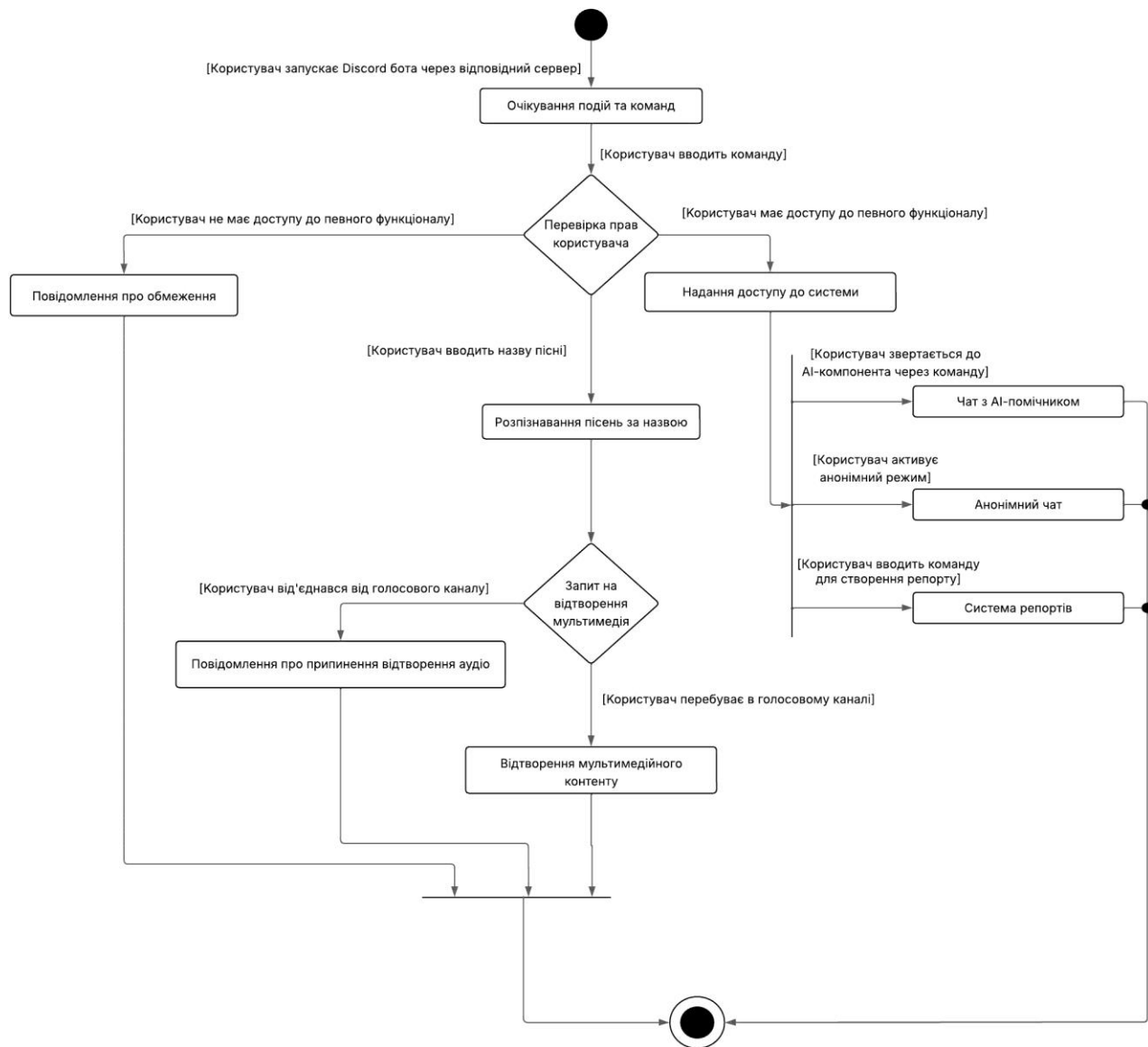


Рисунок 1 – UML-діаграма діяльності Discord бота

На рисунку 2 зображено блок-схему алгоритму реалізації методу динамічного пошуку та керування відтворенням аудіоконтенту. Першим кроком є очікування введення команди від користувача. Бот перевіряє, чи була введена команда /play [4]. Якщо користувач ввів команду неправильно, то бот ігнорує запит і продовжує очікувати введення від користувача. У разі правильного введення команди бот переходить до наступного етапу.

Наступним кроком, користувачу надсилається сповіщення про відтворення аудіо розпачато. Підтвердження цього відправляється до відповідного текстового чату, що дозволяє користувачу переконатися, що його запит був успішно оброблений. Після цього бот приступає до відтворення мультимедійного контенту. У разі відсутності введення користувачем контент вибирається випадковим чином із доступного переліку.

Під час відтворення бот враховує параметри команди, такі як тип контенту та додаткові налаштування, які міг вказати користувач. Також бот аналізує кількість доступних аудіоканалів, щоб забезпечити коректне відтворення без технічних збоїв. Після успішного відтворення аудіо, або у разі

від'єднання від голосового каналу останнього користувача бот припиняє відтворення аудіо. Таким чином, блок-схема демонструє чітку послідовність дій, спрямованих на обробку запиту користувача.

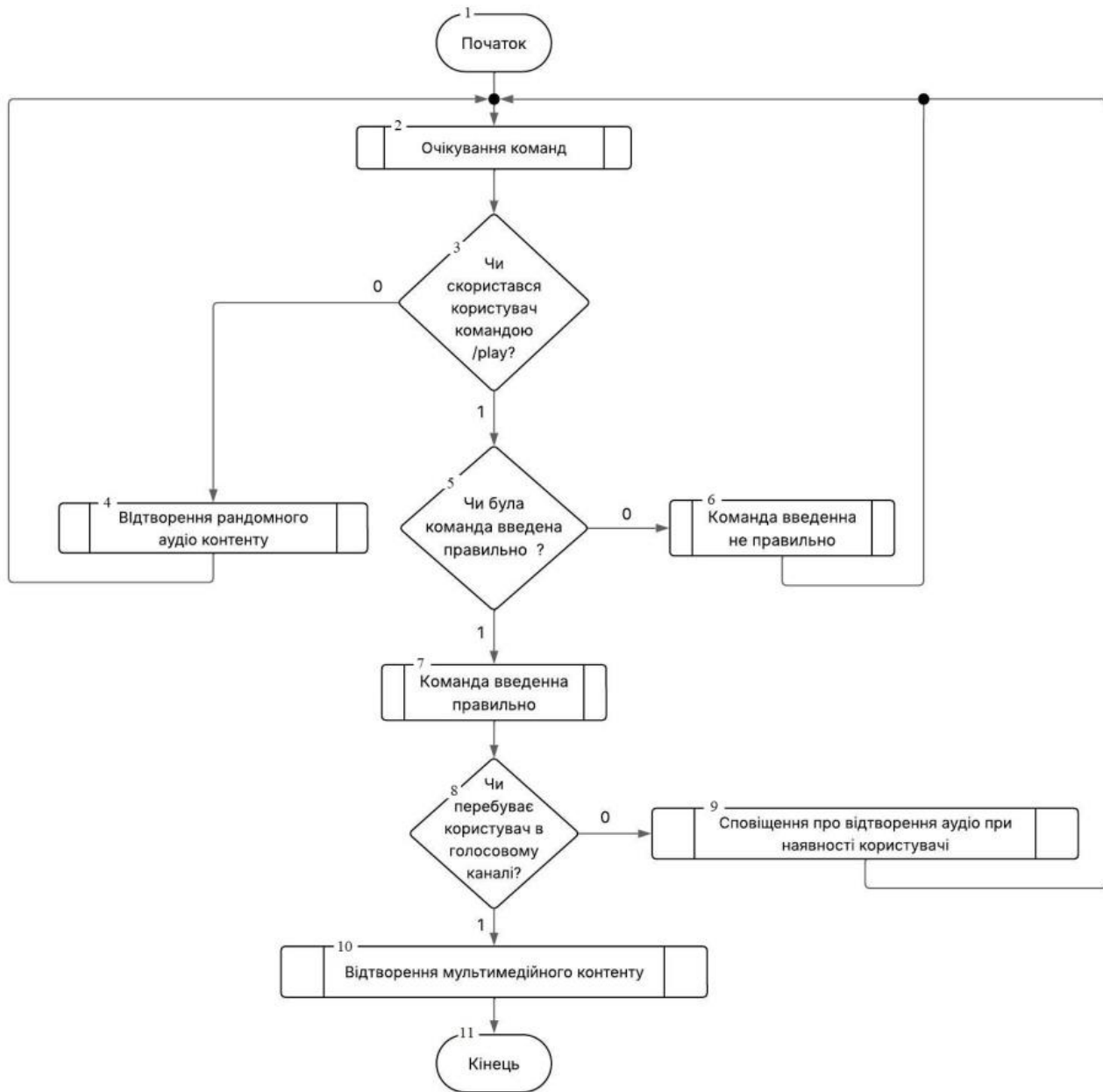


Рисунок 2 – Блок-схема алгоритму реалізації динамічного пошуку та керування відтворенням аудіоконтенту

Також було запропоновано метод модерації контенту, який забезпечує автоматичне виявлення й блокування небажаних, або потенційно шкідливих повідомлень у текстових каналах Discord. Метод базується на поєднанні словникових фільтрів, алгоритмів обробки та шаблонів поведінки користувачів. Інноваційність підходу полягає в оцінці повідомлень із врахуванням історії діалогу, що дозволяє зменшити кількість хибних спрацювань і підвищити ефективність модерації без втручання людини (див рисунок 3).

У разі виявлення серйозних порушень бот може сповістити адміністратора для подальшого розгляду. При умові, якщо контент проходить перевірку, він залишається доступним для інших користувачів, а бот

фіксує успішне завершення модерації. До процесів перевірки можна віднести аналіз контексту ботом та занесення їх у логи для перегляду модерації. Блок-схема демонструє покроковий алгоритм реалізації методу модерації та якості контенту, що є важливим для самого власника та користувачів сервера.

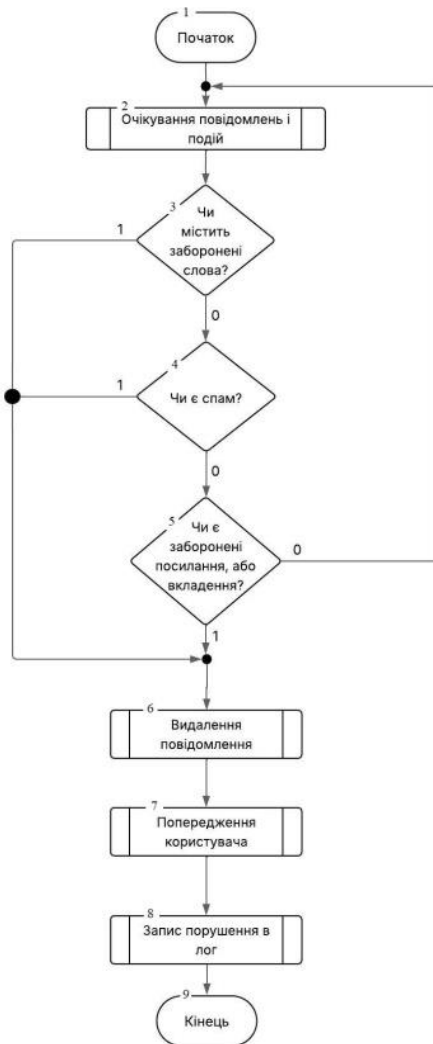


Рисунок 3 – Блок-схема алгоритму модерації контенту

Запропоновані методи стали основою для розробки нового мультимедійного бота для Discord з розширеними можливостями його управління.

Висновок

Таким чином, було розроблено програмне забезпечення мультимедійного бота для Discord, яке поєднує у собі можливості відтворення аудіоконтенту, динамічного пошуку медіафайлів, а також модерації на сервері. Розроблена UML-діаграм діяльності дає чітке уявлення про сценарії використання бота. Для розширення функціональних можливостей бота було розроблено метод динамічного пошуку та керування відтворенням аудіоконтенту, а також удосконалено метод модерації контенту. Це дало можливість підвищити рівень зручності використання для користувачів, мінімізувати кількість помилок та розширити функціональні можливості управління без втручання адміністратора.

СПИСОК ВИКОРИСТАНОЇ ЛІТЕРАТУРИ

1. Романюк О. В., Довгалюк Д. В. Мультимедійний бот для управління Discord каналом. IV Всеукраїнської науково-технічної конференції молодих вчених, аспірантів і студентів «Комп'ютерні ігри і мультимедіа, як інноваційний підхід до комунікації - 2024». м. Одеса, 2024. URL: https://www.ontu.edu.ua/download/konfi/2024/Abstracts_Computer_games_and_multimedia_innovative_approach_electronic_communication_2024.pdf (дата звернення: 10.05.2025).
2. Що таке чат-бот Discord: чи справді він добре працює. URL: <https://www.arkthinker.com/uk/ai-chat/discord-ai-chatbot/> (дата звернення: 10.05.2025).
3. Діаграми UML для моделювання процесів і архітектури. URL: <https://evergreens.com.ua/ua/articles/uml-diagrams.html> (дата звернення: 10.05.2025).
4. Романюк О. В., Довгалюк Д. В. Розробка програмного забезпечення мультимедійного бота для управління Discord каналом. LIV Всеукраїнська науково-технічна конференція підрозділів Вінницького національного технічного університету (2025). Вінниця, 2025. URL: <https://conferences.vntu.edu.ua/index.php/all-fitki/all-fitki-2025/paper/view/24357> (дата звернення: 10.05.2025).

Довгалюк Дмитрій Валентинович – студент групи 4ПІ-21б, факультет інформаційних технологій та комп'ютерної інженерії, Вінницький національний технічний університет, м. Вінниця, e-mail: dimadovgaljuk123@gmail.com

Науковий керівник: **Романюк Оксана Володимирівна**, доцент кафедри програмного забезпечення факультету інформаційних технологій та комп'ютерної інженерії Вінницького національного технічного університету, місто Вінниця. e-mail: romaniukoksana@gmail.com

Dovhaliuk Dmytrii Valentynovych – student of group 4PE-21b, Faculty of Information Technologies and Computer Engineering, Vinnytsia National Technical University, Vinnytsia, e-mail: dimadovgaljuk123@gmail.com

Supervisor: **Romaniuk Oksana Volodymyrivna**, Associate Professor of the Department of Software, Faculty of Information Technology and Computer Engineering, Vinnytsia National Technical University, e-mail: romaniukoksana@gmail.com