

Міністерство освіти і науки України
Криворізький національний університет
Кафедра моделювання та програмного забезпечення

КВАЛІФІКАЦІЙНА РОБОТА
на здобуття ступеня вищої освіти бакалавра
зі спеціальності 121 – Інженерія програмного забезпечення

На тему: Розробка програмної платформи для відтворення потокового мультимедіа

Засвідчую, що в цій
кваліфікаційній роботі немає
запозичень із праць інших
авторів без відповідних
посилань.

Студент гр. ІПЗ–21-2

_____ / В.В. Стуков /

Керівник кваліфікаційної
роботи

_____ / Д. В. Швець /

Завідувач кафедри

_____ / А. М. Стрюк /

Кривий Ріг

2025

Криворізький національний університет

Факультет: Інформаційних технологій

Кафедра: Моделювання та програмного забезпечення

Ступінь вищої освіти: бакалавр

Спеціальність: 121 – Інженерія програмного забезпечення

ЗАТВЕРДЖУЮ:

Зав. кафедри

_____ А. М. Стрюк

«_____» _____ 2025 р.

ЗАВДАННЯ

на кваліфікаційну роботу

студенту групи ІПЗ–21-2 Стукову Валентину Вікторовичу

1. Тема: «Розробка програмної платформи для відтворення потокового мультимедіа» затверджена наказом по КНУ № 200с від «10» квітня 2025 р.
2. Термін подання студентом закінченої роботи: «30» травня 2025 р.
3. Вихідні дані по роботі: розроблений програмний комплекс має забезпечити відтворення потокового аудіо.
4. Зміст пояснювальної записки (перелік питань, що їх треба розробити): провести аналіз існуючих на ринку аналогічних програмних продуктів, обґрунтувати функціонал розроблюваної системи, створити програмне забезпечення для відтворення потокового мультимедіа, здійснити тестування розробленого додатку.
5. Перелік ілюстративного матеріалу: функціональна схема, блок–схема алгоритму, зображення екранних форм додатку.

Календарний план:

№	Найменування етапів кваліфікаційної роботи	Термін виконання етапів роботи
1	Розгляд літературних джерел та пошук інтернет-ресурсів з заданої тематики	12.01.25 – 21.01.25
2	Аналіз існуючих методів вирішення проблеми	22.01.25 – 05.02.25
3	Формулювання актуальності роботи і постановка завдань	06.02.25 – 17.02.25
4	Оформлення матеріалів першого розділу роботи	18.02.25 – 02.03.25
5	Створення функціональної системи та алгоритму додатку	03.03.25 – 15.03.25
6	Оформлення матеріалів другого розділу роботи	16.04.25 – 07.04.25
7	Розробка баз даних, інтерфейсу програмного забезпечення, програмних модулів	08.04.25 – 01.05.25
8	Оформлення додатків	02.05.25 – 07.05.25
9	Тестування розробленої програми	08.05.25 – 15.05.25
10	Оформлення пояснювальної записки	16.05.25 – 29.05.25

Дата видачі завдання: «11» січня 2025 р.

Студент: _____ / В. В. Стуков /

Керівник роботи: _____ / Д. В. Швець /

РЕФЕРАТ

ПОТОКОВЕ МУЛЬТИМЕДІА, VLC, ОНЛАЙН РАДІОМОВЛЕННЯ,
ПОТОКОВЕ АУДІО, ПРОГРАМНЕ ЗАБЕЗПЕЧЕННЯ, ДОДАТОК.

Пояснювальна записка: 60 с., 14 рис., 2 табл., 2 дод., 25 джерел.

Мета кваліфікаційної роботи: розробка програмної платформи для відтворення потокового мультимедіа

Об'єкт проектування: програмна платформа для відтворення потокового мультимедіа.

У теоретичній частині роботи виконано аналіз існуючих на сьогодні аналогічних програмних рішень на ринку. Зазначені сильні та слабкі сторони існуючих програмних продуктів. Обґрунтовані актуальність роботи, мета та сформульовані завдання для розробки зазначеної системи.

У практичній частині кваліфікаційної роботи реалізовано функціональну схему розроблюваного продукту та алгоритм його роботи. Розроблено інтерфейс програмного продукту, програмну логіку роботи додатку. Проведено тестування розробленого програмного забезпечення.

Розроблена програмна платформа для відтворення потокового мультимедіа може використовуватись широким загалом користувачів.

ABSTRACT

**STREAMING MEDIA, VLC, ONLINE RADIO BROADCASTING,
STREAMING AUDIO, SOFTWARE, APPLICATION.**

Explanatory note: 60 p., 14 fig., 2 tabl., 2 app., 25 references.

The aim of the qualifying work: development of a software platform for streaming media playback.

Design object: a software platform for streaming media playback.

In the theoretical part of the work, an analysis of today's similar software decisions in the market have executed. The strengths and weaknesses of existing software products have listed. The urgency of work, the purpose, and the objectives for developing the specified system have formulated.

The functional scheme of the developed product and its algorithm has realized in the practical part of the qualification work. The interface of the software product, the program logic of the application have developed. The developed software has tested.

A wide range of users can use the developed software platform for streaming media playback.

ЗМІСТ

ВСТУП.....	8
1 ОГЛЯД ПИТАННЯ ВИКОРИСТАННЯ ПОТОКОВОГО МУЛЬТИМЕДІА ТА РОЗРОБКИ ЗАСОБІВ ДЛЯ ВІДТВОРЕННЯ АУДІОПОТОКІВ	10
1.1 Загальний огляд питання використання потокового мультимедіа та його актуальності	10
1.2 Короткий історичний огляд становлення технологій передачі потокового мультимедіа	12
1.3 Огляд поняття інтернет-радіомовлення.....	13
1.4 Аналіз існуючих програмних засобів для відтворення потокового мультимедіа.....	14
1.4.1 RarmaRadio	14
1.4.2 RadioZilla	16
1.4.3 Replay Radio	18
1.4.4 TuneIn Radio	20
1.4.5 Radio Garden	22
1.4.6 Audials Play.....	23
1.5 Висновки по проведеному аналізу питання розвитку потокового мультимедіа	26
2 ПРОЕКТУВАННЯ ПРОГРАМНОЇ ПЛАТФОРМИ ДЛЯ ВІДТВОРЕННЯ ПОТОКОВОГО МУЛЬТИМЕДІА	28
2.1 Функціональні та нефункціональні вимоги до розроблюваного програмного забезпечення для відтворення потокового мультимедіа	28
2.2 Поняття якості потокової трансляції.....	28
2.3 Вимоги до обладнання для програмної платформи для відтворення потокового мультимедіа	30

3	РОЗРОБКА ПРОГРАМНОГО ЗАБЕЗПЕЧЕННЯ ДЛЯ ВІДТВОРЕННЯ ПОТОКОВОГО МУЛЬТИМЕДІА.....	31
3.1	Функціональна схема інтернет-радіоприймача	31
3.2	Алгоритм функціонування інтернет-радіоприймача	32
3.3	Використані бібліотеки відтворення потокового мультимедіа для розробки програмного забезпечення	34
3.4	Скріншоти та опис розробленої програмної платформи для відтворення потокового мультимедіа	36
	ВИСНОВКИ	41
	ПЕРЕЛІК ПОСИЛАНЬ	42
	Додаток А	45
	Додаток Б.....	48

ВСТУП

Останні досягнення у сфері цифрових технологій та стрімкий розвиток технічного прогресу суттєво змінили значну кількість аспектів повсякденного життя людини, особливо з урахуванням повсюдного впливу Інтернету. Однією з галузей, що набуває динамічного розвитку, стало потокове мультимедіа, яке швидко еволюціонує завдяки можливостям глобальної мережі. Сучасні технології потокового мультимедіа поступово витісняють традиційні способи передачі інформації, надаючи користувачам доступ до аудіо- та відеоконтенту незалежно від їхнього місцезнаходження. Це відкриває ширші можливості як для споживачів, так і для кріейторів медіаконтенту.

Популярність потокового мультимедіа пояснюється низкою переваг. Користувачі отримують змогу вільно обирати контент відповідно до власних інтересів, не обмежуючись стандартними програмами чи розкладом. Крім того, сучасні мультимедійні сервіси пропонують величезну різноманітність жанрів, форматів і тематик, що дозволяє задовольнити смаки майже всіх користувачів. Гнучкість таких сервісів забезпечує адаптацію під персональні вподобання, а інтерактивність дозволяє керувати процесом відтворення у зручний для користувача спосіб.

Для повноцінного використання можливостей потокового мультимедіа необхідне спеціалізоване програмне забезпечення, яке може бути реалізоване як у вигляді окремих пристроїв, так і у вигляді додатків для персональних комп'ютерів чи смартфонів. Такі програми не лише забезпечують відтворення потокового аудіо чи відео, а й можуть пропонувати додаткові функції, наприклад, персоналізовані рекомендації, інтеграцію з іншими сервісами, збереження улюбленого контенту та інші можливості для підвищення зручності користувача.

В умовах воєнного стану та нестабільного ефірного покриття, питання доступності мультимедійного контенту набуває особливої ваги. У ситуаціях, коли традиційні джерела інформації стають недоступними, саме потокове

мультимедіа може забезпечити оперативний доступ до важливих новин, освітніх чи розважальних матеріалів. Деякі користувачі віддають перевагу аудіо- чи відеоформату для отримання інформації, що часто є найбільш оптимальним рішенням у складних умовах.

Зважаючи на це, розробка програмного забезпечення для відтворення потокового мультимедіа, є актуальним завданням. Особливо це важливо у сучасних реаліях, коли своєчасний доступ до інформації може бути критичним для різних верств населення нашої держави.

1 ОГЛЯД ПИТАННЯ ВИКОРИСТАННЯ ПОТОКОВОГО МУЛЬТИМЕДІА ТА РОЗРОБКИ ЗАСОБІВ ДЛЯ ВІДТВОРЕННЯ АУДІОПОТОКІВ

1.1 Загальний огляд питання використання потокового мультимедіа та його актуальність

Інформаційні технології в XXI сторіччі пронизують усі сфери людської діяльності. Однією з найбільш помітних змін останнього десятиріччя стало масове поширення потокового мультимедіа - технології, яка дозволяє передавати аудіо та відео у режимі реального часу через Інтернет. Ця революція у сфері цифрових комунікацій суттєво змінила уявлення про те, як ми споживаємо інформацію та взаємодіємо з медіаконтентом.

Ще десятиліття тому для прослуховування музики чи перегляду відео потрібно було завантажувати повні файли на пристрій, що займало час і вимагало значних обсягів пам'яті. Сьогодні ж технології потокової передачі даних дозволяють отримувати доступ до масивних мультимедійних ресурсів миттєво, без необхідності їх попереднього завантаження. Це стало можливим завдяки стрімкому розвитку мережевої інфраструктури, збільшенню швидкості інтернет-з'єднань та вдосконаленню алгоритмів стиснення даних.

Серед різноманіття форматів потокового мовлення особливе місце займає інтернет-радіо. Воно поєднує в собі переваги традиційного ефірного радіо, такі як простота використання та можливість фонового прослуховування, з перевагами сучасних цифрових технологій – необмеженим вибором контенту, високою якістю звуку та інтерактивними функціями. На відміну від звичайного радіо, яке обмежене частотами ефірного мовлення, інтернет-радіо не знає кордонів і доступне будь-де, де є підключення до мережі.

Сьогодні майже кожен користувач смартфона або комп'ютера регулярно слухає музику чи подкасти в інтернеті. Це стало можливим завдяки швидкому

розвитку мобільних технологій та вдосконаленню технологій інтернет-з'єднань. Однак разом із зростанням кількості слухачів з'являються й нові складнощі. Наприклад, як забезпечити якісний звук при обмеженій швидкості інтернету? Як обробити одночасне підключення тисяч користувачів? Ці питання потребують професійного підходу та нових рішень.

Існуючі платформи для інтернет-радіо, хоч і є досить поширеними, часто мають свої недоліки. Деякі скаржаться на переривання трансляції, інші – на обмежений вибір радіостанцій або незручний інтерфейс. Крім того, багато радіостанцій стикаються з технічними труднощами при спробі розширити свою аудиторію або впровадити нові формати мовлення.

Важливим аспектом є також економічна доцільність таких проектів. Інтернет-радіо відкриває нові можливості для ведення бізнесу в медіасфері. Це може бути як окремий прибутковий проект, так і додатковий канал комунікації для існуючого бізнесу. Наприклад, музичні лейбли використовують інтернет-радіо для просування своїх виконавців, а великі компанії – для створення корпоративних радіостанцій. Не можна не згадати і про культурний аспект. Інтернет-радіо дозволяє зберігати та поширювати національну музичну спадщину, підтримувати молодих виконавців, об'єднувати людей з різних куточків світу навколо спільних музичних смаків. Воно стає майданчиком для культурного обміну та взаємозбагачення.

Крім того, розробка програмної платформи для інтернет-радіо – це цікаве завдання з технічної точки зору. Воно поєднує в собі роботу з аудіо- та мережевими технологіями, створення зручного інтерфейсу, забезпечення безпеки даних користувачів. Такі проекти стають полем для втілення нових ідей та вдосконалення професійних навичок.

У розрізі сучасних тенденцій до цифровізації та зростання попиту на якісний аудіоконтент, розробка сучасної платформи для інтернет-радіо є актуальною й перспективною сферою діяльності та є важливим об'єктом для практичної реалізації.

1.2 Короткий історичний огляд становлення технологій передачі потокового мультимедіа

Розвиток технологій потокової передачі даних зазнав значної еволюції, яка тісно пов'язана із загальним прогресом комп'ютерних технологій та мережевої інфраструктури. Цей шлях можна умовно поділити на кілька ключових етапів.

Перші кроки у напрямку потокової передачі даних були зроблені ще в 1960-х роках, коли вчені почали експериментувати з передачею аудіо через комп'ютерні мережі. Проте справжній початок потокового мовлення припадає на 1990-ті роки, коли з'явилися перші технології, здатні забезпечити передачу аудіо та відео в режимі реального часу через Інтернет. Важливою віхою стало створення компанією RealNetworks у 1995 році RealAudio [1] - першої широко поширеної технології потокового аудіо.

На початку 2000-х років з'явилися більш досконалі формати стиснення даних, такі як MP3 [2] для аудіо та MPEG-4 [3] для відео, що дозволило значно зменшити обсяг переданих даних без втрати якості. Це сприяло масовому поширенню поточкових сервісів серед широкого загалу користувачів.

Важливим етапом став перехід від пропрієтарних технологій до відкритих стандартів. У 2002 році був затверджений стандарт MPEG-4, а пізніше з'явилися такі технології, як H.264 (AVC) [4] для відео та AAC [5] для аудіо, які стали основою сучасного потокового мовлення.

Поява смартфонів та планшетів у кінці 2000-х років зумовила необхідність адаптації потокового контенту для мобільних пристроїв. Це призвело до розробки адаптивних технологій потокової передачі, таких як HLS (HTTP Live Streaming) [6] від Apple та MPEG-DASH [7], які автоматично регулюють якість потоку залежно від швидкості інтернет-з'єднання.

Окремо варто відзначити вплив технології 5G [8], яка обіцяє зробити якісну поточкову передачу даних доступною практично у будь-якому місці земної кулі.

1.3 Огляд поняття інтернет-радіомовлення

Сучасні інтернет-радіоприймачі відкривають нові можливості для радіомовлення, кардинально відрізняючись від традиційних приймачів своїм принципом роботи. На відміну від звичайних радіоприймачів, які отримують аналоговий сигнал через ефірне мовлення, інтернет-радіоприймачі використовують цифрові технології, підключаючись до мережі через Wi-Fi або Ethernet.



Рисунок 1.1 – Апаратний інтернет-радіоприймач

Якість звучання в інтернет-радіо перевершує традиційне FM-мовлення завдяки використанню цифрових технологій передачі даних. Це особливо помітно при наявності стабільного високошвидкісного інтернет-з'єднання, яке забезпечує чисте та якісне звучання без притаманних аналоговому мовленню перешкод.

Географічна доступність є однією з основних переваг інтернет-радіо. Слухачі отримують можливість підключатися до улюблених радіостанцій з будь-якої точки планети, де є доступ до інтернету, що робить кордони для

радіомовлення неактуальними. Це дозволяє слухачам знаходити радіостанції, що відповідають їхнім музичним смакам та інтересам, серед тисяч доступних варіантів різних мовних та жанрових напрямків.

Сучасні інтернет-радіостанції також часто пропонують персоналізований підхід до кожного слухача. Користувачі можуть налаштовувати потоки музики або новин відповідно до своїх уподобань, обираючи між різними музичними напрямками або мовами мовлення. Багато з цих станцій також пропонують можливість слухати музику без рекламних вставок.

Особливу актуальність інтернет-радіо набуває в умовах воєнного стану. Коли традиційні засоби масової інформації можуть бути обмежені або заблоковані, інтернет-радіо стає надійним джерелом інформації. Воно здатне працювати навіть при частковому руйнуванні інфраструктури, використовуючи супутникові або мобільні мережі для передачі сигналу.

Глобальний характер інтернет-радіо дозволяє поширювати інформацію за межі зони конфлікту, привертаючи увагу міжнародної спільноти до подій, що відбуваються. Крім того, у складних умовах воно може виконувати важливу соціальну функцію, надаючи психологічну підтримку через надання можливості прослуховувати знайомі музичні композиції, новинні огляди та розважальні програми, що допомагає людям долати стресові ситуації та зберігати емоційну стійкість.

1.4 Аналіз існуючих програмних засобів для відтворення потокового мультимедіа

1.4.1 RarmaRadio

RarmaRadio [9], як зазначається на веб-ресурсі розробників даного програмного продукту, дозволяє слухати радіостанції з різних куточків планети та записувати їх.

Скріншот RarmaRadio показано на рисунку 1.2.

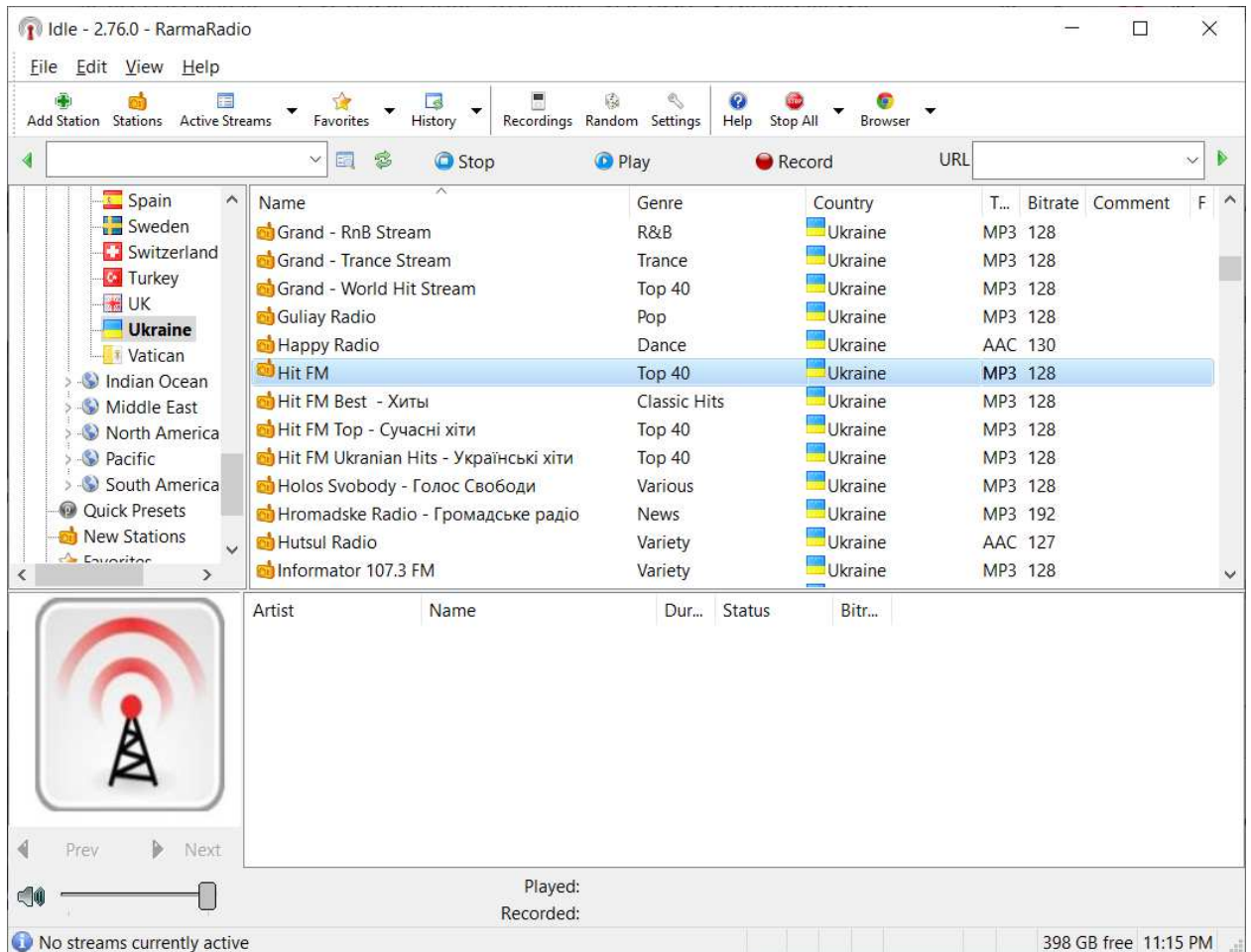
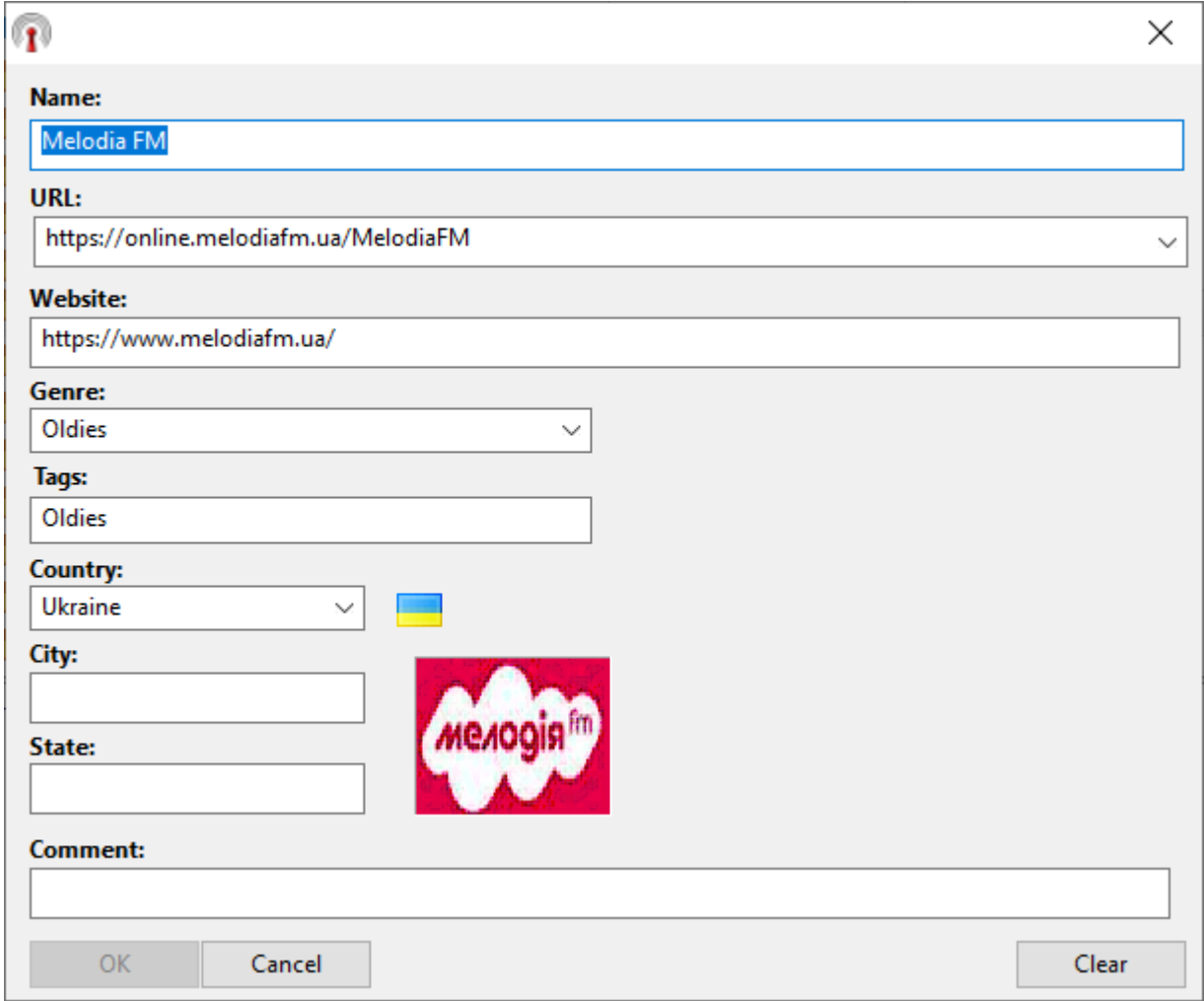


Рисунок 1.2 – RarmaRadio

Ключові функції програмного забезпечення RarmaRadio полягають в наступному:

- прослуховування та запис декількох потоків радіостанцій одночасно;
- автоматичний запис музики за розкладом;
- можливість запису у форматах mp3, wma та ogg;
- відображення обкладинки та тексту композицій;
- безкоштовні оновлення;
- ліцензія на будь-яку кількість персональних комп'ютерів за ціною однієї;
- можливість додавати нові станції або редагувати вже існуючі у базі (рисунок 1.3).



Name:
Melodia FM

URL:
https://online.melodiafm.ua/MelodiaFM

Website:
https://www.melodiafm.ua/

Genre:
Oldies

Tags:
Oldies

Country:
Ukraine

City:

State:

Comment:

OK Cancel Clear

Рисунок 1.3 – RarmaRadio – редагування інтернет-радіостанцій у базі

Програма RarmaRadio є платною та вимагає сплати \$19 за її використання [10]. Інтерфейс додатку виглядає дещо застаріло, хоча останні оновлення RarmaRadio датуються 2025 роком.

1.4.2 RadioZilla

RadioZilla [11] - ще одна безкоштовна програма для Windows і Mac, яка дає змогу не тільки слухати інтернет-радіо на комп'ютері, а й зберігати аудіозаписи на жорсткому диску у форматі MP3. Це означає, що за допомогою цієї програми можна записати пісню або цікаву радіопрограму, трансльовану в прямому ефірі зі станції, на яку налаштований користувач, одним клацанням миші.

Програма проста у використанні, вона пропонує наступний графічний інтерфейс (рисунок 1.4).

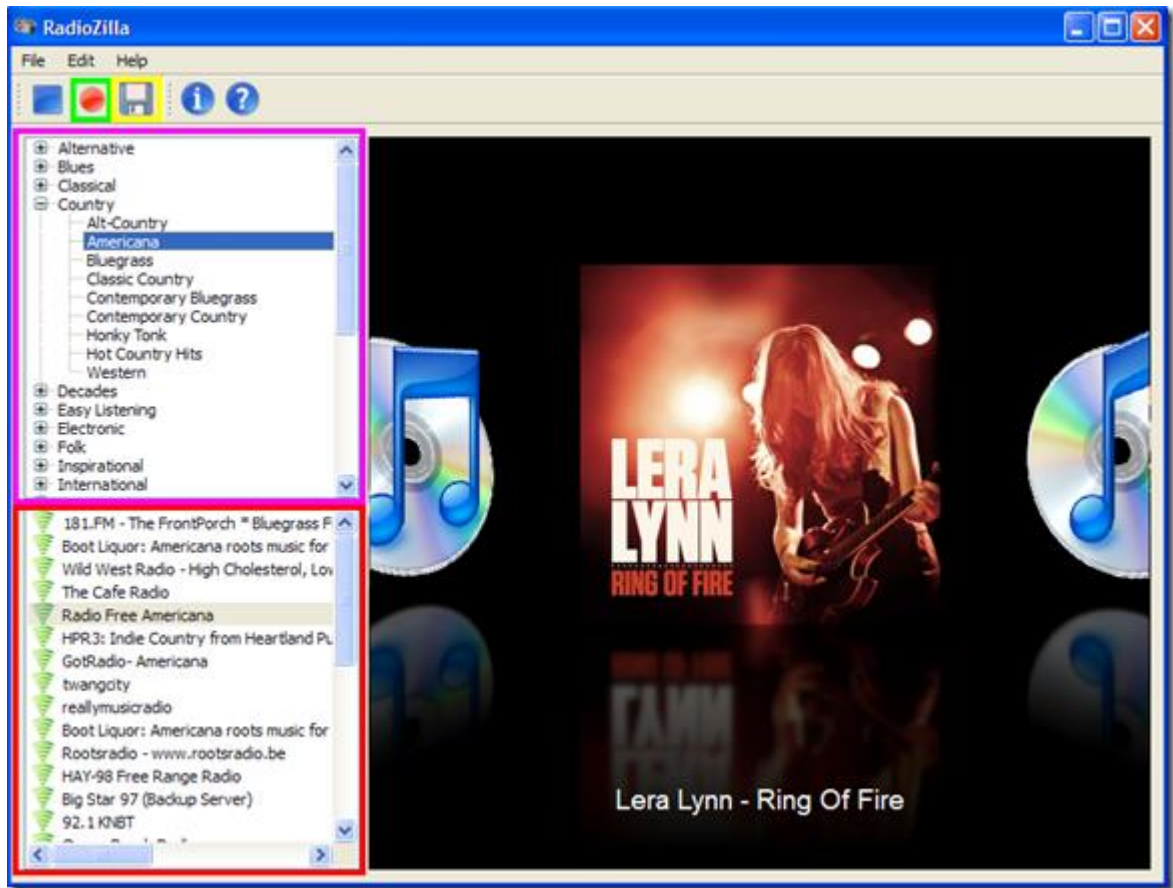


Рисунок 1.4 – RadioZilla

Говорячи про станції, RadioZilla містить достатньо великий архів, і створює користувачеві складнощі, якій зі станцій віддати перевагу. Крім того, для зручності радіостанції розділені та згруповані за жанрами, тож можна відкрити список тільки тих станцій, які грають рок, електроніку, поп, кантрі, фолк, реггі, класику, джаз, ток-шоу та багато іншого.

У колонці в лівій частині екрана спочатку потрібно натиснути на відповідну кнопку +, щоб розгорнути музичний жанр, який цікавить, а потім ще раз натиснути, щоб вибрати один із запропонованих варіантів (підкатегорій).

Відразу ж після цього в червоному полі автоматично відображується список доступних радіостанцій для обраного музичного жанру. Залишається знайти потрібну і вибрати її клацанням миші, щоб почати відтворення. Цікавою особливістю RadioZilla є те, що для деяких пісень, які лунають в ефірі, вона також може відображати обкладинку альбому і назву композиції, яка програватиметься.

Якщо потрібно записати прослуховувану радіопрограму на комп'ютері у вигляді MP3-файлу, треба натиснути на кнопку Rec, виділену зеленим кольором. Якщо потрібно виділити тільки пісню, яка звучить у цей момент, треба натиснути на кнопку, виділену жовтим кольором. Таким чином, RadioZilla завантажить на комп'ютер тільки цю пісню, зберігши її з назвою та ім'ям виконавця або групи, яка її виконує.

Багато користувачів зазначають, що RadioZilla не є одним з найкращих рішень на ринку програмних інтернет-радіоприймачів. Багато, якщо не більшість, радіостанцій банально не працюють. Немає жодних повідомлень про помилки, окрім "не вдається підключитися". Радіостанції також відсортовані не за алфавітом.

Є і проблеми з поділом на категорію – наприклад у розділі "Класика" є станції, які грають класичний рок і романтичні пісні про кохання замість класичної музики.

Загалом, RadioZilla - це непогана задумка із достатньо поганим виконанням. Порівняно з іншими аналогами, вона не є оптимальним варіантом для використання.

1.4.3 Replay Radio

Replay Radio [12] - це ще один спосіб прослуховувати та записувати радіопередачі та подкасти. Вона має широкий спектр функцій. Зовнішній вигляд показано на рисунку 1.5.

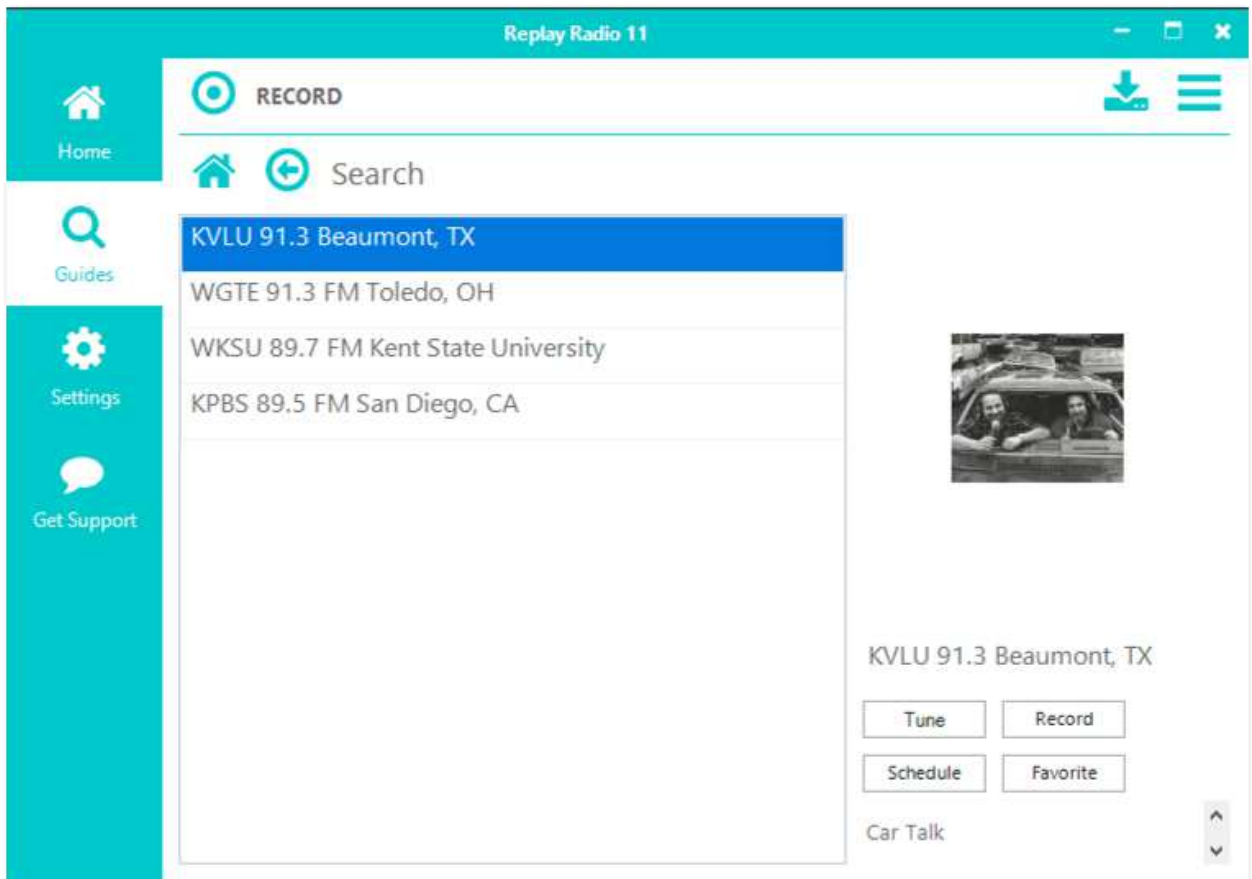


Рисунок 1.5 – Replay Radio

Використання Replay Radio допоможе виконати наступні завдання та пропонує наступні можливості:

- запис Windows Media або будь-які інші формати інтернет-трансляцій;
- інтегрований медіа-довідник відтворення містить понад тисячу радіопередач і понад дві тисячі радіостанцій;
- запис інтернет-радіотрансляцій у форматах MP3, WMA, OGG, WAV та AAC;
- створення аудіо- та MP3-дисків для відтворення у оптичних приводах;
- автоматичний імпорт записаних шоу в iTunes;
- створення закладок для файлів на iPod;

- можливість розділення виконаних записів за тишею, часом або хвилинами;
- зменшення часу буферизації завдяки використанню функції стиснення потоку;
- видалення довгих періодів мовчання із записів (у випадку відсутності мовлення);
- функція швидкого запису дозволяє миттєво створювати записи та зберігати їх;
- інтегрований програвач для відтворення на комп'ютері інтернет-потоку та функція пропуску реклами;
- підтримка бездротових бумбоксів та медіа-подовжувачів через uPnP;
- додатковий покращений драйвер звукової карти забезпечує найвищу якість запису;
- спеціальна система конвертації для додавання власних конвертацій;
- запис з радіоприймачів та інших пристроїв, підключених до комп'ютера;
- запис з мікрофона.

На офіційному сайті Replay Radio зазначено [13], що для використання додатку необхідно придбати річну ліцензію за \$30, що є непомірно дорогого для зазначеного класу програм.

1.4.4 TuneIn Radio

TuneIn [14] - одна з найбільш доступних програм для прослуховування потокового аудіо. Для використання пропонуються мобільні додатки для iOS, Android, включаючи CarPlay, Android Auto і навіть на Android Automotive, який встановлюється на деякі автомобілі заводами-виробниками. Наявні

також і десктопні додатки для Windows PC і MacOS, хоча можна використовувати і веб-браузер.

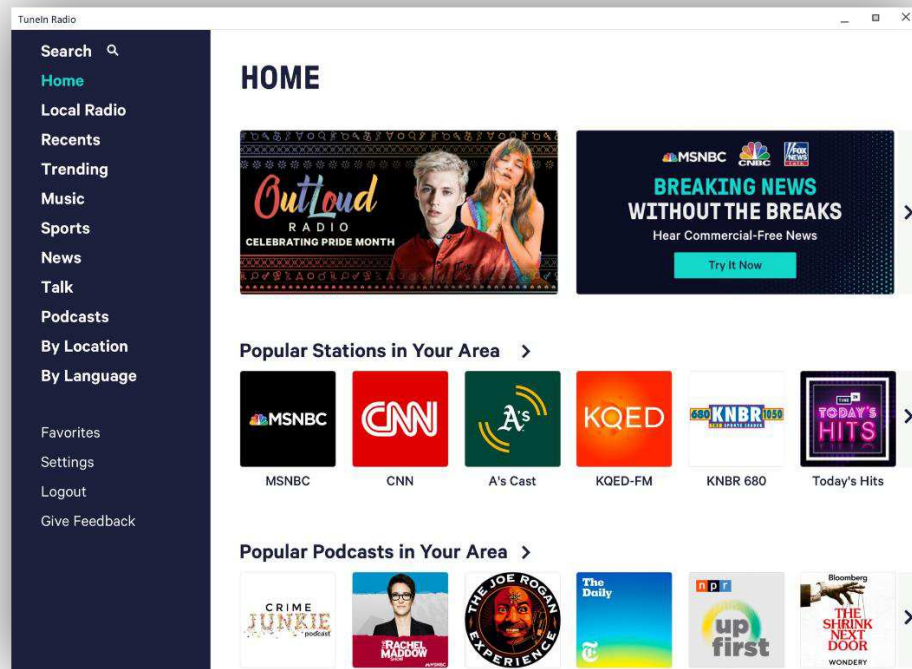


Рисунок 1.6 – TuneIn Radio – версія для операційної системи Windows

TuneIn транслює безліч радіостанцій з усього світу. Останній підрахунок компанії налічує 120 тисяч станцій з 197 країн, які слухають 75 мільйонів слухачів. Додаток розвивається вже протягом майже десятиріччя, тому він і є достатньо розповсюдженим серед користувачів.

Користувачі можуть слухати велику кількість контенту безкоштовно. Не потрібно платити за прослуховування понад сотні тисяч інтернет-радіостанцій і мільйонів подкастів. Розроблено окремі додатки для телефонів і планшетів, а також для операційної системи Windows. TuneIn Radio - це безкоштовна програма, і хоча її завантаження нічого не коштує, вона завантажує і показує рекламу слухачеві.

TuneIn пропонує дві безкоштовні пробні версії для свого преміум-рівня. Можна обрати 7-денну або 30-денну пробну версію, хоча для початку

користувачеві потрібно зареєструватися і ввести свою платіжну інформацію [15].

1.4.5 Radio Garden

Radio Garden [16,17] є некомерційним цифровим і радіодослідницьким проєктом, проведеним Нідерландським інститутом звуку і зору та іншими європейськими університетами в період із 2013 до 2016 року. Основна концепція проєкту - зменшити кордони радіомовлення. У 2016 році проєкт набув масової популярності, подолавши бар'єр у вісім тисяч зареєстрованих радіостанцій.

Зовнішній вигляд Radio Garden наведено на рисунку 1.7. Сайт надає унікальний інтерфейс у вигляді тривимірної карти, де користувачі можуть переміщатися глобусом і прослуховувати місцеві радіостанції, створюючи ефект, подібний до короткохвильових радіоприймачів, але в цьому випадку з використанням технології стримінгу даних. Головна сторінка, названа "Live", дає змогу користувачам досліджувати радіомовлення по всьому світу в режимі реального часу, просто обертаючи земну кулю в своєму браузері. Ця сторінка також надає дані про країну, з якої надходить сигнал.

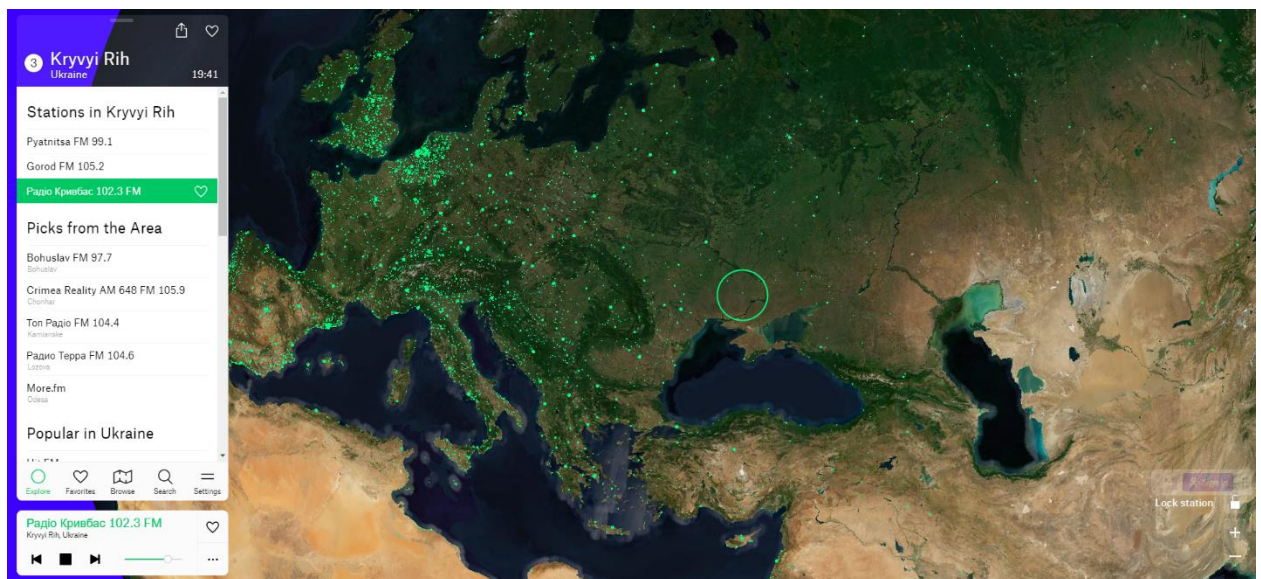


Рисунок 1.7 – Radio Garden

У Radio Garden радіостанції згруповані та впорядковані за геолокацією та містами. Зелені сфери на картографічному відображенні, що збільшуються в міру збільшення кількості радіостанцій у регіоні, забезпечують відповідний оригінальний дизайн сайту. При натисканні на зазначену зелену сфери відбувається розгортання переліку радіостанцій у обраному місті, а також наявні поблизу міста інші радіостанції. Зовнішній вигляд цього переліку наведено на рисунку 1.13.

Сайт використовує загальний домен верхнього рівня .garden, який спочатку призначався для професіоналів у сфері садівництва, і, як зазначається на спеціалізованих сайтах, його інтерфейс є адаптивним і підходить для будь-якого браузера та роздільної здатності. Для передачі сигнал, що генерується мовником, повинен бути перетворений з радіоформату у потоковий формат. Підтримувані формати перетворення для потокового мовлення - MP3, OGG і AAC.

В цілому, розробка є оригінальним рішенням порівняно з іншими засобами прослуховування інтернет-радіостанцій, та заслуговує на увагу. Втім, при використанні зазначеного сайту спостерігалось використання значних обсягів трафіку, що зумовлено високим бітрейтом трансляцій [18].

1.4.6 Audials Play

Audials Play [19] позиціонується як зручний і безкоштовний додаток для прослуховування інтернет-радіо, який надає доступ до понад ста тисяч радіостанцій і тисяч подкастів різними мовами. Audials Play дозволяє шукати, слухати, обирати та записувати музику, радіопередачі, новини та трансляції спортивних подій.

Вигляд програми Audials Play для Windows наведено на рисунку 1.8.

Серед функцій Audials Play можна виділити наступний достатньо широкий перелік:

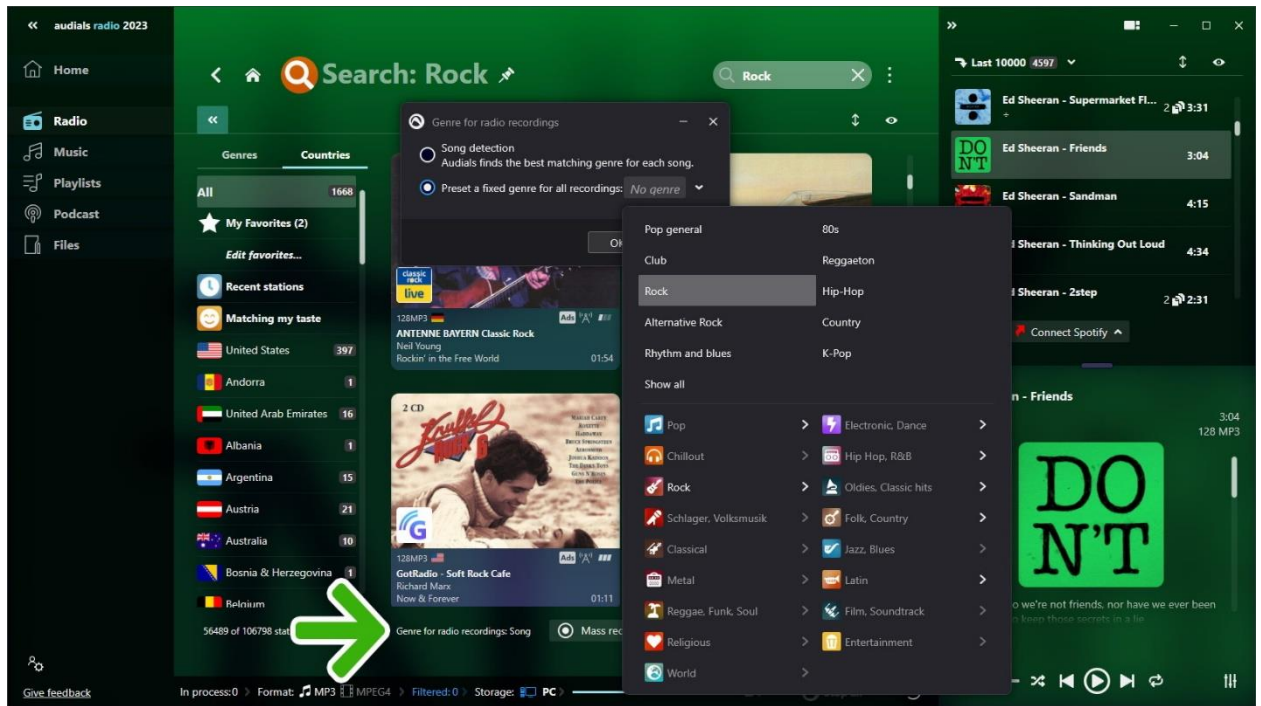


Рисунок 1.8 – Audials Play – версія для операційної системи Windows

- Бібліотека з понад ста тисяч станцій - Audials Radio надає чіткий доступ до радіостанцій з усього світу. Користувач може обирати з 200 жанрів та незліченної кількості країн і регіонів;
- Запис пісень з радіопотоків з високою якістю. Можливість використання таймеру запису та оптимізації записаної музики. Audials може одночасно записувати до 100 станцій;
- Автоматичне додавання обкладинок і текстів пісень до кожного запису та можливість зміни тегів вручну;
- Створення списків з улюбленими станціями та виконавцями. Audials формує релевантними пропозиції, що базується на попередніх прослуховуваннях користувача;
- Широкий вибір музики - завдяки нескінченному вибору музики та актуальних виконавців, треків і хіт-парадів користувач завжди матиме найкращу музику на всі випадки життя. Завдяки функції "Музичний зум" можна ще більше зануритися у безмежний світ музики;

- Точний аудіоредактор - редагування музики, додавання затухань на початку або в кінці треків, нормалізація гучності та експорт фрагментів пісні;
- Автоматичне додавання тегів;
- Експорт музики на пристрої користувача - Audials впорядковує улюблені композиції користувача і передає їх на комп'ютер, смартфон або в хмару;
- Керування хмарним сховищем і завантаженням музики на смартфон;
- Виявлення, редагування та видалення дублікатів пісень;
- Понад два мільйони подкастів з найсвіжішим контентом: новини, спорт, комедії, гостросюжетні фільми і т. д.;
- Зручна каталогізація - понад 100 організованих категорій та потужний повнотекстовий пошук;
- Гарантії повернення грошей за купівлю повної версії у разі, якщо сервіс не сподобається користувачеві.

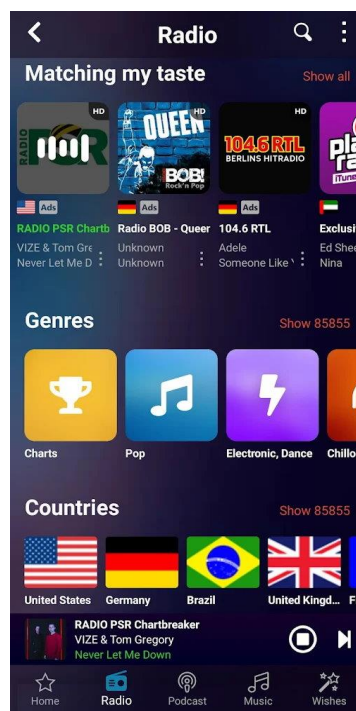


Рисунок 1.9 – Audials Play – додаток для Android

Audials Play може використовуватись на різних пристроях і гаджетах під управлінням різних операційних систем.

1.5 Висновки по проведеному аналізу питання розвитку потокового мультимедіа

Результати дослідження інтернет-джерел свідчать про те, що радіомовлення залишається ефективним засобом масової інформації завдяки ряду істотних переваг. Однією з головних особливостей радіо є його здатність супроводжувати слухача в різних життєвих ситуаціях. На відміну від інших медіа, які вимагають повної уваги, радіо дозволяє отримувати інформацію паралельно з виконанням повсякденних справ, будь то приготування їжі, керування автомобілем або робота. Ця якість робить його невід'ємною частиною повсякденного життя великої кількості людей.

Важливою перевагою радіомовлення є його менша наповненість рекламним контентом порівняно з іншими медіаплатформами, зокрема телебаченням. Це створює більш комфортні умови для сприйняття інформації, оскільки слухачі не відчують себе перевантаженими надмірною кількістю рекламних повідомлень.

Унікальність радіо як засобу масової інформації полягає також у його здатності стимулювати уяву слухачів. На відміну від телебачення або інтернет-контенту, які надають готові картинки, радіо змушує слухача активізувати власну уяву, створюючи уявні образи на основі почуті інформації. Це робить процес сприйняття більш особистим і глибоким.

Розвиток цифрових технологій відкриває нові можливості для радіомовлення. Інтернет-радіо обходить географічні кордони, дозволяючи слухачам долучатися до ефірів радіостанцій з усього світу. Це особливо актуально для українських слухачів, які можуть мати обмежений доступ до національного мовлення на тимчасово окупованих територіях або в умовах блекаутів.

У контексті енергетичних обмежень особливого значення набуває питання ефективного використання інтернет-трафіку. Розробка програмного забезпечення для інтернет-радіо повинна враховувати необхідність мінімізації споживання трафіку, що дозволить забезпечити стабільний доступ до радіомовлення навіть за умови обмеженої пропускної здатності каналу зв'язку. Цей аспект стає особливо актуальним у надзвичайних ситуаціях, коли надійний доступ до інформації є критично важливим.

2 ПРОЕКТУВАННЯ ПРОГРАМНОЇ ПЛАТФОРМИ ДЛЯ ВІДТВОРЕННЯ ПОТОКОВОГО МУЛЬТИМЕДІА

2.1 Функціональні та нефункціональні вимоги до розроблюваного програмного забезпечення для відтворення потокового мультимедіа

Програма розробляється в якості інтернет-радіоприймача, який має дозволити слухати радіостанції з усього світу, забезпечуючи при цьому високу якість звуку та стабільність роботи.

Додаток має забезпечити підтримку відтворення радіостанцій з різних куточків світу, включаючи наступні країни та регіони:

- Україна;
- Східно-Європейські країни (Польща, Румунія і т.д.);
- Велика Британія;
- Північно-американський регіон (Сполучені Штати Америки та Канада);
- Австралія та Океанія;

Інтерфейс користувача має бути простим та інтуїтивно зрозумілим, при цьому надаючи швидкий доступ до всіх необхідних функцій. Основні елементи керування мають бути розташовані зручно та логічно, що має дозволити керувати програмою в кілька кліків.

Також додаток має згортатися в системний трей, де він може продовжувати працювати, не захаращуючи екран. При цьому має бути забезпечена можливість призупинення трансляції та перемикання радіостанцій шляхом виконання кліків по значку програми в системному треї.

2.2 Поняття якості потокової трансляції

Ключовим поняттям в контексті якості відтворення інтернет аудіо-потoku є бітрейт.

Бітрейт означає кількість біт на секунду (біт/с), яка може бути передана цифровою мережею. Іншими словами, це міра "пропускної здатності" або швидкості передачі даних інтернет-з'єднання. Це дуже важливо для інтернет-радіо, оскільки безпосередньо впливає на якість потокового аудіо [20,21].

Кілька важливих моментів, які слід розглянути, говорячи про бітрейт трансляції:

- Якість - вищий бітрейт зазвичай означає вищу якість звуку. Це пояснюється тим, що для захоплення звуку використовується більше даних. Наприклад, аудіопотік зі швидкістю 320 кбіт/с звучатиме краще, ніж потік зі швидкістю 128 кбіт/с, оскільки для відтворення оригінального звуку використовується більше даних;

- Використання пропускної здатності - вищий бітрейт означає, що мережею передається більше даних, що вимагає більшої пропускної здатності. Якщо пропускна здатність мережі обмежена, високий бітрейт може спричинити буферизацію або переривання. Тому інтернет-радіо часто надають різні варіанти бітрейту, щоб пристосуватись до різних швидкостей інтернету в користувачів;

- Розмір файлу - вищий бітрейт також означає більший розмір файлу. Це важливо, якщо слухач має лімітоване з'єднання, де потрібно економити використання даних. Саме це є центральним питанням даної роботи;

- Сумісність - не всі пристрої або інтернет-з'єднання можуть працювати з високими бітрейтами. Деякі старі пристрої можуть не відтворювати їх. Аналогічно, повільніші інтернет-з'єднання можуть не встигати за потоками з високим бітрейтом.

Бітрейт для інтернет-радіо зазвичай коливається від 64 кбіт/с до 320 кбіт/с (високоякісне аудіо). "Золота середина" для багатьох інтернет-радіостанцій становить близько 128 кбіт/с, що забезпечує хороший баланс між якістю звуку та використанням даних.

Таким чином, бітрейт інтернет-трансляції визначальним чином впливає на споживання трафіку при роботі програмної платформи для відтворення потокового мультимедіа та, відповідно, зумовлює якість звуку.

2.3 Вимоги до обладнання для програмної платформи для відтворення потокового мультимедіа

В таблиці 2.1 описані вимоги, які мають бути задовільнені для ефективної роботи програмної платформи для відтворення потокового мультимедіа

Таблиця 2.1 – Вимоги до апаратної складової

Характеристика	Вимоги
Процесор (CPU)	2,0 GHz
Оперативна пам'ять (RAM)	1,5 ГБ
Місце на жорсткому диску (HDD/SSD)	30 МБ
Інтернет-з'єднання	+
Монітор	Роздільна здатність 800*600
Клавіатура та миша	+
Принтер	Ні

Слід зазначити, що наявність інтернет-підключення є основною вимогою, хоча його якість не є основним критерієм.

3 РОЗРОБКА ПРОГРАМНОГО ЗАБЕЗПЕЧЕННЯ ДЛЯ ВІДТВОРЕННЯ ПОТОКОВОГО МУЛЬТИМЕДІА

3.1 Функціональна схема інтернет-радіоприймача

Функціональну схему програмного забезпечення для відтворення потокового мультимедіа наведено на рисунку 3.1.

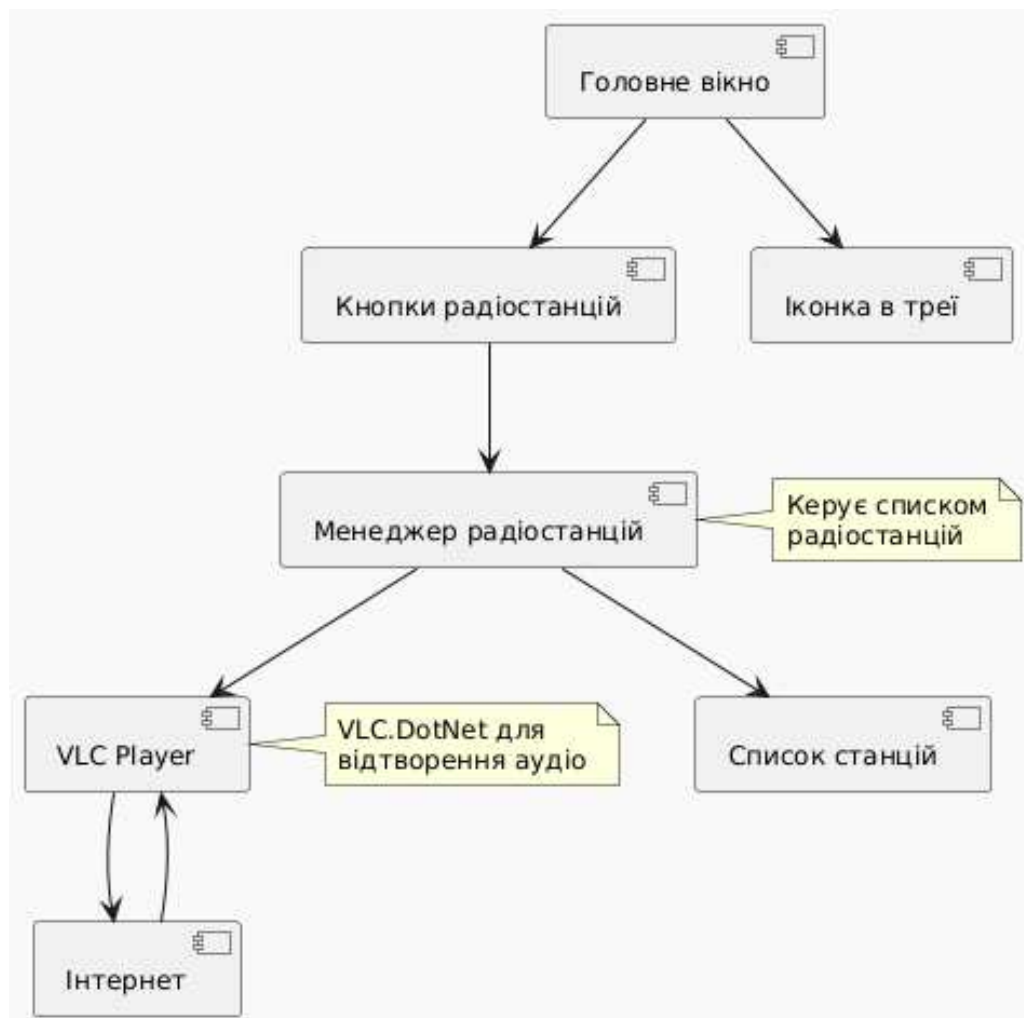


Рисунок 3.1 – Функціональна схема інтернет-радіоприймача

Як видно з зображення, функціональна схема інтернет-радіоприймача включає в себе такі логічні блоки, як «Головне вікно», «Кнопки радіостанцій», «Менеджер радіостанцій», «Іконка в треї», «Список станцій», «Інтернет» та «VLC Player».

Основою логіки програми є VLC Player, який відповідає за відтворення аудіопотоків. Він використовує бібліотеку VLC.DotNet, що забезпечує високу стабільність роботи та підтримку різноманітних аудіоформатів. Цей компонент дозволяє програмам .NET інтегрувати можливості медіаплеєра VLC, надаючи доступ до широкого спектру кодеків і протоколів.

Користувацький інтерфейс програми організований навколо головного вікна, яке слугує центральною точкою взаємодії. Воно забезпечує доступ до всіх основних функцій через інтуїтивно зрозумілі елементи керування.

Важливу роль у програмі відіграє менеджер радіостанцій, який виступає посередником між інтерфейсом користувача та аудіоплеєром. Він відповідає за завантаження списку доступних станцій, обробку вибору користувача та керування станом відтворення.

Архітектура програми передбачає можливість легкого розширення функціоналу. Це стосується як додавання нових джерел аудіо, так і впровадження додаткових можливостей, таких як запис ефіру, створення плейлистів або інтеграція зі сторонніми сервісами. Це забезпечує можливість розвитку програми у відповідь на потенційну зміну вимог користувачів.

3.2 Алгоритм функціонування інтернет-радіоприймача

На рисунку 3.2 наведено алгоритм роботи розроблюваного інтернет-радіоприймача.

Процес роботи програми починається з ініціалізації та завантаження списку доступних радіостанцій з локальної бази даних. Це дозволяє забезпечити миттєвий доступ до переліку станцій навіть за відсутності інтернет-з'єднання.

Після завантаження списку користувач отримує змогу обрати бажану радіостанцію через інтуїтивно зрозумілий інтерфейс. Вибір здійснюється за допомогою натискання на потрібну кнопку користувачем. Після вибору станції програма автоматично встановлює з'єднання з відповідним аудіопотоком.

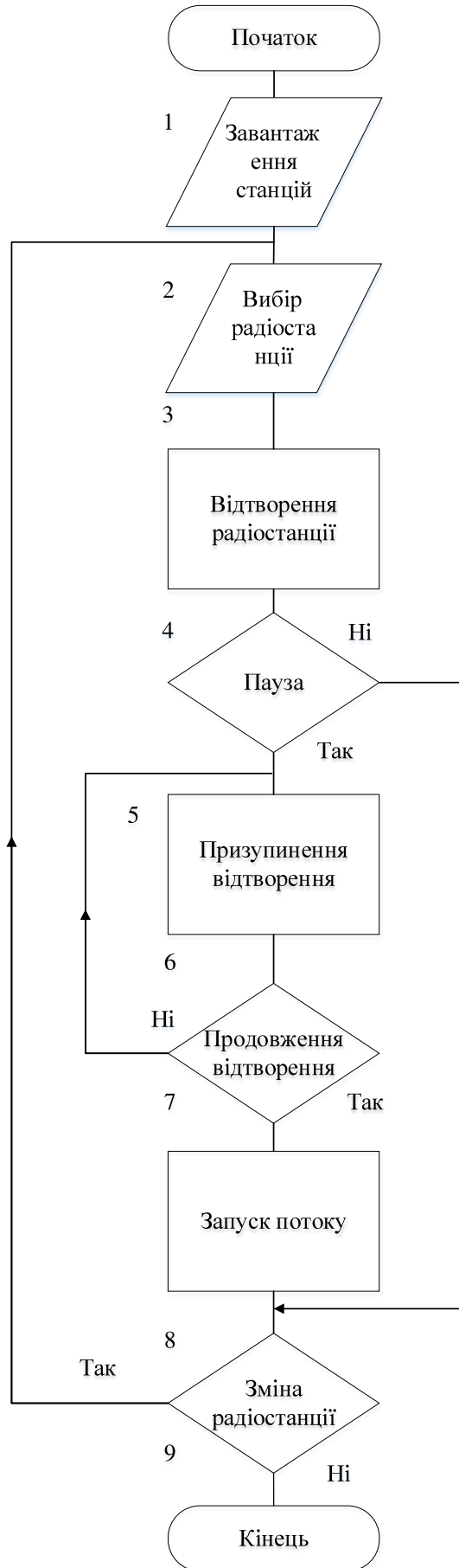


Рисунок 3.2 – Алгоритм роботи інтернет-радіоприймача

Користувач має можливість у будь-який момент змінити радіостанцію, обравши іншу шляхом вибору відповідної кнопки. При цьому програма автоматично перериває поточне відтворення та ініціює підключення до обраної станції.

Для зручності роботи запропоновано згортання програми в трей, а також призупинення відтворення та запуск випадкової радіостанції з переліку при натисканні на іконку в треї.

Запропонований алгоритм роботи забезпечує стабільне функціонування інтернет-радіоприймача.

3.3 Використані бібліотеки відтворення потокового мультимедіа для розробки програмного забезпечення

За для забезпечення реалізації відтворення аудіопотоків було застосовано бібліотеку VLC.DotNet [22].

VLC.DotNet є інструментом для розробників .NET [23], який надає доступ до можливостей бібліотеки VLC через .NET інтерфейс. Ця бібліотека надає можливості до створення професійних аудіододатків, починаючи від простих програвачів аудіофайлів різних форматів до складних систем інтернет-радіомовлення.

Особливістю VLC.DotNet є його універсальність у роботі з різними аудіоформатами. Він легко справляється з відтворенням як популярних форматів на кшталт MP3 чи AAC, так і більш специфічних аудіокодеків, що робить його зручним вибором для розробників, допомагаючи забезпечити широку підтримку аудіоформатів у своїх додатках.

Важливою перевагою бібліотеки є її здатність ефективно працювати з мережевими потоками. Вона добре обробляє різноманітні мережеві протоколи, дозволяючи програмам безпосередньо підключатися до поточкових джерел. Ця можливість стає в нагоді при створенні інтернет-радіо, оскільки дозволяє уникнути необхідності реалізації складного мережевого коду.

Робота з бібліотекою відрізняється високим рівнем надійності. Вона має вбудовані механізми обробки помилок, які автоматично впораються з нестабільними з'єднаннями, намагаючись відновити перервані потоки без втручання користувача. Це рішення добре підходить для застосунків, де важлива безперервність аудіомовлення.

Архітектура бібліотеки дозволяє ефективно керувати системними ресурсами, що робить її придатною для використання як у легких клієнтських додатках, так і у більш складних системах. Підтримка асинхронних операцій забезпечує плавну роботу інтерфейсу користувача, запобігаючи його блокуванню під час операцій з мережею.

Інтеграція з графічними інтерфейсами Windows Forms [24] або WPF [25] відбувається інтуїтивно зрозуміло, дозволяючи розробникам створювати сучасні та естетичні інтерфейси для керування відтворенням. Додатковою перевагою є можливість застосування різноманітних аудіофільтрів, які дозволяють покращити якість звучання або адаптувати його під уподобання користувача.

Хоча в даному випадку розглядається Windows-версія, слід зазначити, що VLC.DotNet підтримує крос-платформеність, що відкриває перспективи для майбутнього розширення можливостей додатку. Бібліотека гармонійно поєднується з іншими інструментами розробки, що дозволяє створювати комплексні рішення для обробки та відтворення аудіоконтенту.

Далі наведено перелік основних форматів, відтворення яких підтримується бібліотекою VLC.DotNet.

MP3 (MPEG-1 Audio Layer III) став синонімом цифрового аудіо завдяки своїй здатності значно зменшувати розмір аудіофайлів при збереженні прийнятної якості звучання. Формат використовує втратне стиснення, видаляючи звуки, які важко почути людському вуху. Незважаючи на появу новіших форматів, MP3 залишається популярним через універсальну підтримку та сумісність з більшістю пристроїв та програмного забезпечення.

Типовий бітрейт для інтернет-радіо складає 128-192 кбіт/с, що забезпечує гарний баланс між якістю звуку та витратами трафіку.

AAC (Advanced Audio Coding) був розроблений як наступник MP3 та пропонує кращу якість звучання при однакових бітрейтах. Він ефективніше обробляє високі частоти та забезпечує більш чіткий звук у порівнянні з MP3. AAC став стандартом для багатьох сучасних сервісів, включаючи iTunes, YouTube та інші потокові платформи. Формат підтримує різні профілі, що дозволяє адаптувати якість звуку під конкретні вимоги. Для інтернет-радіо часто використовується бітрейт 96-128 кбіт/с, що забезпечує якісне звучання при помірному використанні трафіку.

HLS (HTTP Live Streaming) є поточковим протоколом, розробленим компанією Apple, який став промисловим стандартом для передачі аудіо та відео через Інтернет. Він працює шляхом розділення потоку на невеликі сегменти (зазвичай тривалістю 6-10 секунд), які завантажуються через звичайні HTTP-з'єднання. Це рішення має кілька переваг:

- краща адаптація до змін якості мережі;
- менша ймовірність переривання потоку;
- підтримка шифрування та захисту контенту;
- сумісність з більшістю сучасних пристроїв.

HLS використовує аудіокодеки AAC або HE-AAC для стиснення звуку, поєднуючи переваги ефективного кодування з надійністю поточної передачі. Цей підхід ефективний для трансляцій у прямому ефірі та масштабованих поточкових сервісів.

3.4 Скріншоти та опис розробленої програмної платформи для відтворення поточкового мультимедіа

Зовнішній вигляд розробленої системи для відтворення поточкового аудіо наведено на рисунку 3.3.

Як видно на скріншоті, майже усі кнопки співпадають за розташуванням з числовими значеннями на шкалі, що робить інтуїтивно зрозумілим та зручним вибір радіостанцій користувачем.

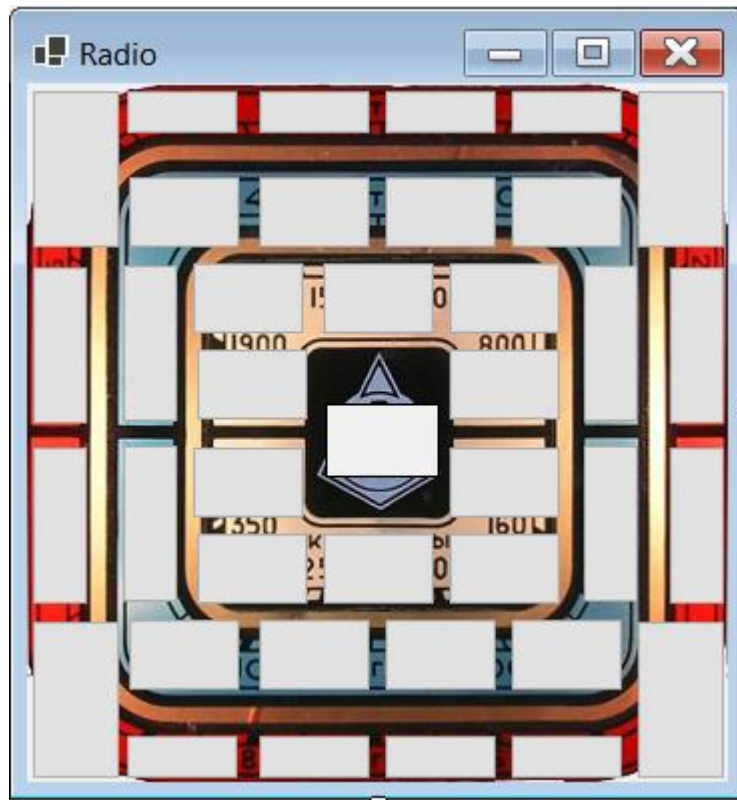


Рисунок 3.4 – Макет, що демонструє розташування елементів керування

Що стосується концепції вибору радіостанцій та їх джерел онлайн-потоків, то тут, в першу чергу, робився нахил на українські станції – як розмовні (новинні), так і музично-розважальні. З іншого боку, необхідно було охопити й джерела інформації з інших країн, що зумовило додавання в перелік достатньої кількості англійських радіостанцій. Їх використання надає користувачеві змогу як дізнаватися міжнародні новини, так і вдосконалювати навички володіння англійською мовою. Також в перелік додано декілька станцій з сусідніх країн (Польща, Румунія).

Розподіл інтернет-радіостанцій за логічними блоками наведено на рисунку 3.5.

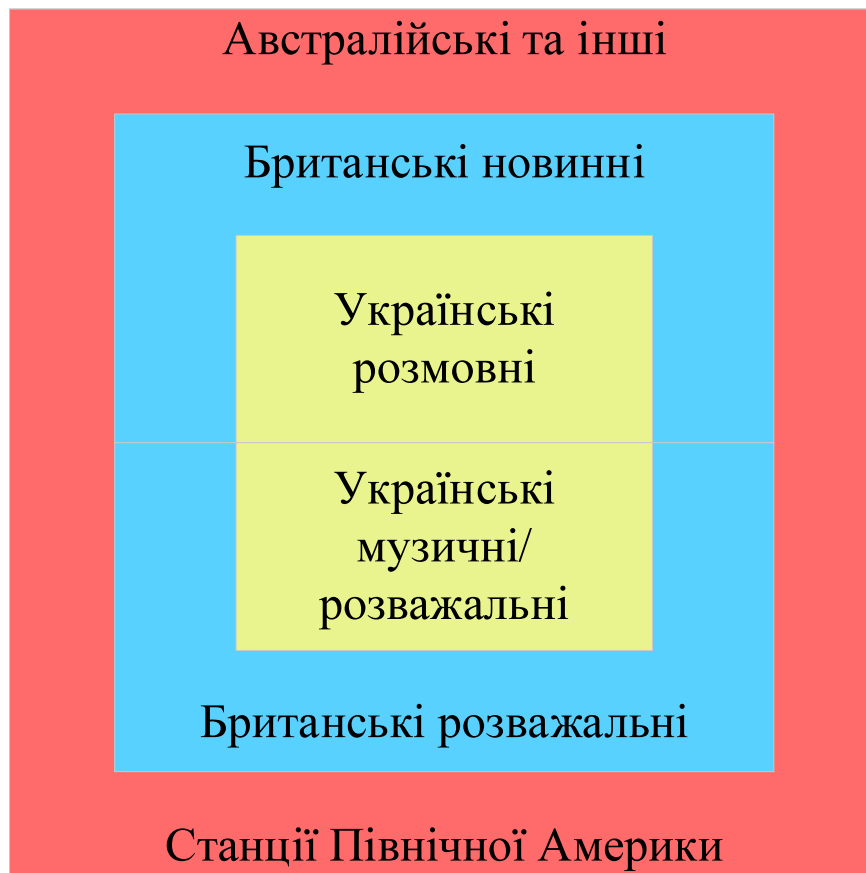


Рисунок 3.5 – Розподіл інтернет-радіостанцій за логічними блоками

Розглянемо логічні блоки детально:

а) *Українські розмовні* – цей блок містить вітчизняні новинні радіостанції, зокрема UA:Суспільне, Радіо NV та інші. Здебільшого орієнтовані на новини, розмовні передачі, ток-шоу, хоча і включає декілька станцій культурно-музичного спрямування;

б) *Українські музичні/розважальні* – вітчизняні радіостанції на кшталт LuxFM, АрміяFM, HitFM, які переважну частину ефірного часу транслюють музичний контент;

в) *Британські новинні* – включають в себе лінійку розмовних каналів від BBC Radio, що здебільшого орієнтуються на новини та передачі розмовного жанру, спортивні новини, невеликою мірою охоплюють культурні вистави та музичні трансляції;

г) *Британські розважальні* – цей розділ представлено каналами компанії PumpkinFM, що пропонують трансляції ретро-записів британського радіо

минулого сторіччя, та включають в себе пригодницькі, детективні, комедійні та інші передачі. З урахуванням того, що більшість записів трансляцій датовано 30-ми та 40-ми роками минулого сторіччя, якість відтворення відповідає тим часам;

д) *Австралійські та інші* – блок представлений радіостанціями тихоокеанського регіону, а також включає в себе поодинокі станції інших країн;

е) *Станції Північної Америки* – включають різного роду англомовні станції Сполучених Штатів та Канади. Серед них культурні, спортивні, новинні радіостанції тощо.

Повний перелік використаних радіостанцій наведено в додатку А.

ВИСНОВКИ

В ході виконання дипломної роботи було розроблено сучасний інтернет-радіоприймач, який поєднує в собі зручний інтерфейс та стабільну роботу. Отримані результати підтверджують ефективність обраного підходу до проектування та реалізації програмного забезпечення для потокового аудіомовлення.

Розроблена програма демонструє високу ефективність у роботі з різними форматами аудіопотоків, забезпечуючи стабільне відтворення навіть при низькій швидкості інтернет-з'єднання. Використання бібліотеки VLC.DotNet дозволило реалізувати заявлений функціонал при мінімальних вимогах до системних ресурсів.

Важливим досягненням є створення інтуїтивно зрозумілого інтерфейсу, який робить програму доступною для користувачів. Можливість роботи у фоновому режимі та інтеграція з системним треем значно підвищують зручність використання програми.

Розробка підтвердила актуальність обраної теми, оскільки інтернет-радіо залишається популярним джерелом інформації та розваг, особливо в умовах, коли традиційне радіомовлення може бути обмеженим або недоступним. Запропоноване рішення може бути використане як самостійний продукт, так і як основа для більш складних систем потокового мовлення.

Перспективи подальшого розвитку проекту включають розширення списку підтримуваних форматів, реалізації функціоналу для додавання власних радіостанцій, а також впровадження механізмів для персоналізації станцій на основі вподобань користувача.

Зазначений інтернет-радіоприймач може використовуватись широким загалом користувачів.

ПЕРЕЛІК ПОСИЛАНЬ

1. RealMedia — мультимедійний формат, розроблений RealNetworks [Електронний ресурс] – Режим доступу до ресурсу: <https://uk.wikipedia.org/wiki/RealMedia>
2. MP3 [Електронний ресурс] – Режим доступу до ресурсу: <https://en.wikipedia.org/wiki/MP3>
3. MPEG-4 [Електронний ресурс] – Режим доступу до ресурсу: <https://en.wikipedia.org/wiki/MPEG-4>
4. Що таке H.264, H.264+, H.265, H.265+? [Електронний ресурс] – Режим доступу до ресурсу: <https://control.ua/ua/blog/chto-takoe-h-264-h-264-h-265-h-265.html>
5. AAC (розширене кодування звуку) : що це таке? [Електронний ресурс] – Режим доступу до ресурсу: <https://gsmhub.com.ua/glossary/aac-rozsirene-koduvannia-zvuku>
6. Що таке пряма трансляція HLS HTTP? [Електронний ресурс] – Режим доступу до ресурсу: <https://mimolive.com/uk/hls-http-%D0%BF%D1%80%D1%8F%D0%BC%D0%B0-%D1%82%D1%80%D0%B0%D0%BD%D1%81%D0%BB%D1%8F%D1%86%D1%96%D1%8F/>
7. What is MPEG-DASH and How Does Dynamic Adaptive Streaming Compare to HLS? [Електронний ресурс] – Режим доступу до ресурсу: <https://getstream.io/glossary/mpeg-dash-streaming/>
8. What is 5G? [Електронний ресурс] – Режим доступу до ресурсу: https://aws.amazon.com/what-is/5g/?nc1=h_ls
9. RarmaRadio 2.0. Bringing Internet Radio To Life. [Електронний ресурс] – Режим доступу до ресурсу: <http://raimersoft.com/rarmaradio.html>
10. RarmaRadio Buy Online [Електронний ресурс] – Режим доступу до ресурсу: http://raimersoft.com/buy_rarmaradio.html

11. RadioZilla [Електронний ресурс] – Режим доступу до ресурсу:
<http://www.softsea.com/review/RadioZilla.html>
12. Replay Radio – the best way to enjoy your favorite live radio shows and stations [Електронний ресурс] – Режим доступу до ресурсу:
<https://www.applian.com/replay-radio/>
13. Subscribe to Replay Radio [Електронний ресурс] – Режим доступу до ресурсу: https://applian.com/order/order_RRAD?
14. TuneIn [Електронний ресурс] – Режим доступу до ресурсу:
<https://tunein.com/>
15. Want to listen to TuneIn Radio? What you need to know [Електронний ресурс] – Режим доступу до ресурсу: <https://getconnectedmedia.com/want-to-listen-to-tunein-radio-what-you-need-to-know/>
16. Radio Garden [Електронний ресурс] – Режим доступу до ресурсу:
<https://radio.garden/>
17. Radio Garden Andriod App [Електронний ресурс] – Режим доступу до ресурсу:
<https://play.google.com/store/apps/details?id=com.jonathanpuckey.radiogarden&hl=ru&gl=US>
18. Radio Garden [Електронний ресурс] – Режим доступу до ресурсу:
https://uk.wikipedia.org/wiki/Radio_Garden
19. Audials Play [Електронний ресурс] – Режим доступу до ресурсу:
<https://audials.com/en/audials-radio>
20. Бітрейт [Електронний ресурс] – Режим доступу до ресурсу:
<https://vue.gov.ua/Бітрейт>
21. Бітова швидкість [Електронний ресурс] – Режим доступу до ресурсу:
https://uk.wikipedia.org/wiki/Бітова_швидкість
22. .NET control that hosts the audio/video capabilities of the VLC libraries [Електронний ресурс] – Режим доступу до ресурсу:
<https://github.com/ZeBobo5/Vlc.DotNet>

23. Build. Test. Deploy - .NET is the free, open-source, cross-platform framework for building modern apps and powerful cloud services. [Электронный ресурс] – Режим доступа до ресурсу: <https://dotnet.microsoft.com/en-us/>

24. Windows Forms Overview [Электронный ресурс] – Режим доступа до ресурсу: <https://learn.microsoft.com/en-au/dotnet/desktop/winforms/overview/>

25. Windows Presentation Foundation Overview [Электронный ресурс] – Режим доступа до ресурсу: <https://learn.microsoft.com/en-au/dotnet/desktop/wpf/overview/>

Додаток А

Перелік використаних радіостанцій

Таблиця А.1 – Станції та джерела потокового мультимедіа

Назва станції	Країна	URL
Українське радіо	Україна	http://radio.ukr.radio:8000/ur1-aacp-ulow
Радіо Промінь	Україна	http://radio.ukr.radio:8000/ur2-aacp-ulow
Радіо Культура	Україна	http://radio.ukr.radio:8000/ur3-aacp-ulow
Всесвітня служба радіо України	Україна	https://radio.ukr.radio/rui-aacplus-l
Радіо Точка	Україна	https://radio.ukr.radio/ur5-mp3
Радіо НВ	Україна	http://91.218.212.84:8000/radionv.mp3
Мелодія FM	Україна	https://online.melodiafm.ua/MelodiaFM
HitFM	Україна	https://online.hitfm.ua/HitFM
ЛюксFM	Україна	http://icecast.luxnet.ua/luxlviv
АрміяFM	Україна	http://212.26.132.129:8000/ArmyFM
BBC Radio 1	Велика Британія	https://as-hls-ww-live.cf.md.bbc.co.uk/pool_904/live/ww/bbc_radio_one/bbc_radio_one.isml/bbc_radio_one-audio=96000.norewind.m3u8
BBC Radio 3	Велика Британія	https://as-hls-ww-live.akamaized.net/pool_904/live/ww/bbc_radio_three/bbc_radio_three.isml/bbc_radio_three-audio=96000.norewind.m3u8
BBC Radio 4	Велика Британія	https://a.files.bbc.co.uk/ms6/live/3441A116-B12E-4D2F-ACA8-C1984642FA4B/audio/simulcast/hls/nonuk/audio_

		syndication_low_sbr_v1/cfs/bbc_radio_fourfm.m3u8
BBC Radio 4 Extra	Велика Британія	https://a.files.bbc.co.uk/ms6/live/3441A116-B12E-4D2F-ACA8-C1984642FA4B/audio/simulcast/hls/nonuk/audio_syndication_low_sbr_v1/cfs/bbc_radio_four_extra.m3u8
BBC Radio 5 Sports	Велика Британія	https://a.files.bbc.co.uk/ms6/live/3441A116-B12E-4D2F-ACA8-C1984642FA4B/audio/simulcast/hls/nonuk/audio_syndication_low_sbr_v1/cfs/bbc_radio_five_live.m3u8
BBC Radio 6 Music	Велика Британія	https://as-hls-ww.live.cf.md.bbc.co.uk/pool_904/live/ww/bbc_6music/bbc_6music.isml/bbc_6music-audio=96000.norewind.m3u8
BBC World Service	Велика Британія	http://stream.live.vc.bbcmedia.co.uk/bbc_world_service
Pumpkin FM 1940s Radio	Велика Британія	https://cast2.asurahosting.com/proxy/1940sradio/stream
Pumpkin FM American Gold	Велика Британія	https://cast2.asurahosting.com/proxy/americangold/stream
Pumpkin FM Strange Radio	Велика Британія	https://cast2.asurahosting.com/proxy/strangeradio/stream
Pumpkin FM Venture Radio	Велика Британія	https://cast2.asurahosting.com/proxy/ventureradio/stream
Pumpkin FM Radio England	Велика Британія	https://cast2.asurahosting.com/proxy/radioengland/stream

ABC Triple J NSW Australia	Австралія	https://live- radio02.mediahubaustralia.com/2TJW/mp3
Nau FM	Папуа Нова Гвінея	https://playerservices.streamtheworld.com/api/live stream-redirect/NAUFM.mp3
ABC News Radio	Австралія	https://live- radio02.mediahubaustralia.com/RAEW/aac/;
2RPH 1224 AM Sydney	Австралія	http://27.100.36.118:8000/stream/1/
ABC Radio National NSW	Австралія	https://live- radio02.mediahubaustralia.com/2RNW/mp3
RNZ National	Нова Зеландія	https://radionz-ice.streamguys.com/national.mp3
Antena Satelor	Румунія	http://89.238.227.6:8356
Tok FM	Польща	https://radiostream.pl/tuba10-1.mp3
CBC Radio 1 Ontario	Канада	https://playerservices.streamtheworld.com/api/live stream-redirect/CBQTFM_CBC.mp3
Radio Rebelde	Куба	https://icecast.teveo.cu/zrXXWK9F
Radio Costa	Мексика	https://streaming.shoutcast.com/xhld-radio-costa
NPR News and Talk	США	https://npr-ice.streamguys1.com/live.aac
CNN International	США	https://tunein.streamguys1.com/CNNi
Air 1 Radio	США	https://maestro.emfcdn.com/stream_for/air1/tunein /aac
KFAX AM	США	https://playerservices.streamtheworld.com/api/live stream-redirect/KFAXAM_SC
KKFI	США	https://stream.pacificaservice.org:9000/kkfi_128

Додаток Б

Програмний код

```
using System;
using System.Collections.Generic;
using System.Drawing;
using System.IO;
using System.Linq;
using System.Windows.Forms;
using Vlc.DotNet.Forms;

namespace RadioGadget
{
    public partial class Form1 : Form
    {
        // Плеєр для відтворення аудіо
        private VlcControl vlcControl;

        private bool _isDragging = false;
        private Point _offset;

        // Іконка в трєї
        private NotifyIcon trayIcon;

        public Form1()
        {
            InitializeComponent();

            buttonLW1.Tag = "https://radio.ukr.radio/url-dp-mp3-
m";
            buttonLW2.Tag = "https://radio.ukr.radio/ur2-mp3-m";
            buttonLW3.Tag = "https://radio.ukr.radio/ur3-mp3-m";
            buttonLW4.Tag = "https://radio.ukr.radio/rui-aacplus-
l";
            buttonLW5.Tag = "https://radio.ukr.radio/ur5-mp3";
            buttonLW6.Tag = "https://online-
radio.nv.ua/radionv.mp3";
            buttonLW7.Tag =
"https://online.melodiafm.ua/MelodiaFM";
            buttonLW8.Tag = "https://online.hitfm.ua/HitFM";
            buttonLW9.Tag = "http://icecast.luxnet.ua/luxlviv";
            buttonLW10.Tag = "http://212.26.132.129:8000/ArmyFM";

            buttonMW1.Tag = "https://as-hls-
ww.live.cf.md.bbc.co.uk/pool_904/live/ww/bbc_radio_one/bbc_radi
o_one.isml/bbc_radio_one-audio=96000.norewind.m3u8";
            buttonMW2.Tag = "https://as-hls-ww-
live.akamaized.net/pool_904/live/ww/bbc_radio_three/bbc_radio_th
ree.isml/bbc_radio_three-audio=96000.norewind.m3u8";
```

```
        buttonMW3.Tag =
"https://a.files.bbc.co.uk/ms6/live/3441A116-B12E-4D2F-ACA8-
C1984642FA4B/audio/simulcast/hls/nonuk/audio_syndication_low_sbr
_v1/cfs/bbc_radio_fourfm.m3u8";
        buttonMW4.Tag =
"https://a.files.bbc.co.uk/ms6/live/3441A116-B12E-4D2F-ACA8-
C1984642FA4B/audio/simulcast/hls/nonuk/audio_syndication_low_sbr
_v1/cfs/bbc_radio_four_extra.m3u8";
        buttonMW5.Tag =
"https://a.files.bbc.co.uk/ms6/live/3441A116-B12E-4D2F-ACA8-
C1984642FA4B/audio/simulcast/hls/nonuk/audio_syndication_low_sbr
_v1/cfs/bbc_radio_five_live.m3u8";
        buttonMW6.Tag = "https://as-hls-
ww.live.cf.md.bbc.co.uk/pool_904/live/ww/bbc_6music/bbc_6music.
isml/bbc_6music-audio=96000.norewind.m3u8";
        buttonMW7.Tag =
"http://stream.live.vc.bbcmedia.co.uk/bbc_world_service";
        buttonMW8.Tag =
"https://cast2.asurahosting.com/proxy/1940sradio/stream";
        buttonMW9.Tag =
"https://cast2.asurahosting.com/proxy/americangold/stream";
        buttonMW10.Tag =
"https://cast2.asurahosting.com/proxy/strangeradio/stream";
        buttonMW11.Tag =
"https://cast2.asurahosting.com/proxy/ventureradio/stream";
        buttonMW12.Tag =
"https://cast2.asurahosting.com/proxy/radioengland/stream";

        buttonSW1.Tag = "https://live-
radio02.mediahubaustralia.com/2TJW/mp3";
        buttonSW2.Tag =
"https://playerservices.streamtheworld.com/api/livestream-
redirect/NAUFM.mp3";
        buttonSW3.Tag = "https://live-
radio02.mediahubaustralia.com/RAEW/aac/";
        buttonSW4.Tag =
"http://27.100.36.118:8000/stream/1/";
        buttonSW5.Tag = "https://live-
radio02.mediahubaustralia.com/2RNW/mp3";
        buttonSW6.Tag = "https://radionz-
ice.streamguys.com/national.mp3";
        buttonSW7.Tag = "http://89.238.227.6:8356";
        buttonSW8.Tag = "https://radiostream.pl/tuba10-
1.mp3";
        buttonSW9.Tag =
"https://playerservices.streamtheworld.com/api/livestream-
redirect/CBQTFM_CBC.mp3";
        buttonSW10.Tag = "https://icecast.teveo.cu/zrXXWK9F";
        buttonSW11.Tag =
"https://streaming.shoutcast.com/xhld-radio-costa";
        buttonSW12.Tag = "https://npr-
ice.streamguys1.com/live.aac";
```

```

        buttonSW13.Tag =
"https://tunein.streamguys1.com/CNNi";
        buttonSW14.Tag = "http://online.svitle.org:6728/fm";
        buttonSW15.Tag =
"https://playerservices.streamtheworld.com/api/livestream-
redirect/KFAXAM_SC";
        buttonSW16.Tag =
"https://stream.pacificaservice.org:9000/kkfi_128";

```

```

        // opacity
        SetButtonsTransparency();

```

```

        InitializeVlc();

```

```

        SetFormTransparency();

```

```

        // переміщення форми
        this.MouseDown += Form1_MouseDown;
        this.MouseMove += Form1_MouseMove;
        this.MouseUp += Form1_MouseUp;

```

```

        InitializeTrayIcon();

```

```

        this.ShowInTaskbar = false;

```

```

        this.KeyPreview = true;
        this.KeyDown += Form1_KeyDown;

```

```

    }

```

```

private List<Button> GetAllButtons(Control control)
{

```

```

    var buttons = new List<Button>();

```

```

    foreach (Control c in control.Controls)
    {

```

```

        if (c is Button button)
        {

```

```

            buttons.Add(button);
        }

```

```

        else
        {

```

```

            buttons.AddRange(GetAllButtons(c));
        }
    }

```

```

    return buttons;
}

```

```

private void TrayIcon_MouseDown(object sender,
MouseEventArgs e)
{

```

```

        if (e.Button == MouseButton.Right)
        {
            var buttons = GetAllButtons(this)
                .Where(b => b != buttonStop &&
!string.IsNullOrEmpty(b.Tag?.ToString()))
                .ToList();

            if (buttons.Any())
            {
                var randomButton = buttons[new
Random().Next(buttons.Count)];
                randomButton.PerformClick();
            }
        }
    }

    private void Form1_KeyDown(object sender, KeyEventArgs
e)
    {
        if (e.Alt && e.KeyCode == Keys.F4)
        {
            Application.Exit();
        }
    }

    private void TrayIcon_MouseClick(object sender,
MouseEventArgs e)
    {
        if (e.Button == MouseButton.Left)
        {
            this.Show();
            this.WindowState = FormWindowState.Normal;
            this.Activate(); // Активируем окно
        }
    }

    private void TrayIcon_DoubleClick(object sender,
EventArgs e)
    {
        if (vlcControl != null)
        {
            if (vlcControl.IsPlaying)
            {
                vlcControl.Pause();
            }
            else
            {
                vlcControl.Play();
            }
        }
    }
}

```

```

private void InitializeTrayIcon()
{
    // Створення іконки в треї
    trayIcon = new NotifyIcon();
    trayIcon.Text = "RadioGadget";
    trayIcon.Icon = new Icon("radio-icon.ico");
    trayIcon.Visible = true;

    trayIcon.MouseClick += TrayIcon_MouseClick;
    trayIcon.MouseDown += TrayIcon_MouseDown;
    trayIcon.DoubleClick += TrayIcon_DoubleClick;

    trayIcon.ContextMenuStrip = null;

    this.FormClosing += Form1_FormClosing;
}

private void OnRestore(object sender, EventArgs e)
{
    this.Show();
    this.WindowState = FormWindowState.Normal;
    this.Activate(); // Активіруємо вікно
}

private void OnExit(object sender, EventArgs e)
{
    trayIcon.Visible = false; // Сховуємо іконку в треї
    Application.Exit();
}

private void Form1_FormClosing(object sender,
FormClosingEventArgs e)
{
    if (e.CloseReason == CloseReason.UserClosing)
    {
        e.Cancel = true;
        this.Hide();
        trayIcon.Visible = true;
    }
}

private void SetFormTransparency()
{
    this.FormBorderStyle = FormBorderStyle.None;

    this.BackColor = Color.Magenta;
    this.TransparencyKey = Color.Magenta;

    this.BackgroundImage = Image.FromFile("back1.png");
}

```



```
        this.BackgroundImageLayout = ImageLayout.Stretch;
    }
```

```
private void SetButtonsTransparency()
{
    // Кнопки LW
    SetButtonTransparency(buttonLW1);
    SetButtonTransparency(buttonLW2);
    SetButtonTransparency(buttonLW3);
    SetButtonTransparency(buttonLW4);
    SetButtonTransparency(buttonLW5);
    SetButtonTransparency(buttonLW6);
    SetButtonTransparency(buttonLW7);
    SetButtonTransparency(buttonLW8);
    SetButtonTransparency(buttonLW9);
    SetButtonTransparency(buttonLW10);
    SetButtonTransparency(buttonStop);

    // Кнопки MW
    SetButtonTransparency(buttonMW1);
    SetButtonTransparency(buttonMW2);
    SetButtonTransparency(buttonMW3);
    SetButtonTransparency(buttonMW4);
    SetButtonTransparency(buttonMW5);
    SetButtonTransparency(buttonMW6);
    SetButtonTransparency(buttonMW7);
    SetButtonTransparency(buttonMW8);
    SetButtonTransparency(buttonMW9);
    SetButtonTransparency(buttonMW10);
    SetButtonTransparency(buttonMW11);
    SetButtonTransparency(buttonMW12);

    // Кнопки SW
    SetButtonTransparency(buttonSW1);
    SetButtonTransparency(buttonSW2);
    SetButtonTransparency(buttonSW3);
    SetButtonTransparency(buttonSW4);
    SetButtonTransparency(buttonSW5);
    SetButtonTransparency(buttonSW6);
    SetButtonTransparency(buttonSW7);
    SetButtonTransparency(buttonSW8);
    SetButtonTransparency(buttonSW9);
    SetButtonTransparency(buttonSW10);
    SetButtonTransparency(buttonSW11);
    SetButtonTransparency(buttonSW12);
    SetButtonTransparency(buttonSW13);
    SetButtonTransparency(buttonSW14);
    SetButtonTransparency(buttonSW15);
    SetButtonTransparency(buttonSW16);
}
```

```

private void SetButtonTransparency(Button button)
{
    if (button != null)
    {
        button.FlatStyle = FlatStyle.Flat;
        button.BackColor = Color.Transparent;

        button.FlatAppearance.BorderSize = 0;
        button.FlatAppearance.MouseDownBackColor =
Color.Transparent;
        button.FlatAppearance.MouseOverBackColor =
Color.Transparent;
    }
}

private void InitializeVlc()
{
    try
    {
        var vlcLibDirectory = new
DirectoryInfo(@"C:\Program Files\VideoLAN\VLC");

        if (!vlcLibDirectory.Exists)
        {
            MessageBox.Show("Шлях до VLC не знайдено: "
+ vlcLibDirectory.FullName);

            return;
        }

        // Ініціалізація VlcControl
        vlcControl = new VlcControl();
        vlcControl.BeginInit();
        vlcControl.VlcLibDirectory = vlcLibDirectory;
        vlcControl.Dock = DockStyle.Fill;
        vlcControl.EndInit();
        this.Controls.Add(vlcControl);
        vlcControl.Visible = false;
    }
    catch (Exception ex)
    {
        MessageBox.Show("Помилка ініціалізації VLC: " +
ex.Message);
    }
}

private void buttonLW1_Click(object sender, EventArgs e)
{
    PlayRadio(buttonLW1.Tag.ToString());
}

```

```

private void buttonLW2_Click(object sender, EventArgs e)
{
    PlayRadio(buttonLW2.Tag.ToString());
}

private void buttonLW3_Click(object sender, EventArgs e)
{
    PlayRadio(buttonLW3.Tag.ToString());
}

private void buttonLW4_Click(object sender, EventArgs e)
{
    PlayRadio(buttonLW4.Tag.ToString());
}

private void buttonLW5_Click(object sender, EventArgs e)
{
    PlayRadio(buttonLW5.Tag.ToString());
}

private void buttonLW6_Click(object sender, EventArgs e)
{
    PlayRadio(buttonLW6.Tag.ToString());
}

private void buttonLW7_Click(object sender, EventArgs e)
{
    PlayRadio(buttonLW7.Tag.ToString());
}

private void buttonLW8_Click(object sender, EventArgs e)
{
    PlayRadio(buttonLW8.Tag.ToString());
}

private void buttonLW9_Click(object sender, EventArgs e)
{
    PlayRadio(buttonLW9.Tag.ToString());
}

private void buttonLW10_Click(object sender, EventArgs
e)
{
    PlayRadio(buttonLW10.Tag.ToString());
}

private void buttonStop_Click(object sender, EventArgs
e)
{
    StopRadio();
}

```

```

private void PlayRadio(string url)
{
    if (string.IsNullOrEmpty(url)) return;

    try
    {

        StopRadio();

        vlcControl.SetMedia(new Uri(url));
        vlcControl.Play();

    }
    catch (Exception ex)
    {

    }
}

private void StopRadio()
{
    if (vlcControl != null)
    {
        vlcControl.Stop();
    }
}

private void Form1_MouseDown(object sender,
MouseEventArgs e)

{
    if (e.Button == MouseButtons.Left)
    {
        _isDragging = true;
        _offset = new Point(e.X, e.Y);
    }
}

private void Form1_MouseMove(object sender,
MouseEventArgs e)

{
    if (_isDragging)
    {

        Point newLocation = this.PointToScreen(new
Point(e.X, e.Y));

        newLocation.Offset(-_offset.X, -_offset.Y);
    }
}

```

```
        this.Location = newLocation;
    }
}

e) private void Form1_MouseUp(object sender, MouseEventArgs
{
    if (e.Button == MouseButton.Left)
    {
        _isDragging = false;
    }
}

private void buttonMW1_Click(object sender, EventArgs e)
{
    PlayRadio(buttonMW1.Tag.ToString());
}

private void buttonMW2_Click(object sender, EventArgs e)
{
    PlayRadio(buttonMW2.Tag.ToString());
}

private void buttonMW3_Click(object sender, EventArgs e)
{
    PlayRadio(buttonMW3.Tag.ToString());
}

private void buttonMW4_Click(object sender, EventArgs e)
{
    PlayRadio(buttonMW4.Tag.ToString());
}

private void buttonMW5_Click(object sender, EventArgs e)
{
    PlayRadio(buttonMW5.Tag.ToString());
}

private void buttonMW6_Click(object sender, EventArgs e)
{
    PlayRadio(buttonMW6.Tag.ToString());
}

private void buttonMW7_Click(object sender, EventArgs e)
{
    PlayRadio(buttonMW7.Tag.ToString());
}

private void buttonMW8_Click(object sender, EventArgs e)
{
    PlayRadio(buttonMW8.Tag.ToString());
}
```

```
private void buttonMW9_Click(object sender, EventArgs e)
{
    PlayRadio(buttonMW9.Tag.ToString());
}

e) private void buttonMW10_Click(object sender, EventArgs
{
    PlayRadio(buttonMW10.Tag.ToString());
}

e) private void buttonMW11_Click(object sender, EventArgs
{
    PlayRadio(buttonMW11.Tag.ToString());
}

e) private void buttonMW12_Click(object sender, EventArgs
{
    PlayRadio(buttonMW12.Tag.ToString());
}

private void buttonSW1_Click(object sender, EventArgs e)
{
    PlayRadio(buttonSW1.Tag.ToString());
}

private void buttonSW2_Click(object sender, EventArgs e)
{
    PlayRadio(buttonSW2.Tag.ToString());
}

private void buttonSW3_Click(object sender, EventArgs e)
{
    PlayRadio(buttonSW3.Tag.ToString());
}

private void buttonSW4_Click(object sender, EventArgs e)
{
    PlayRadio(buttonSW4.Tag.ToString());
}

private void buttonSW5_Click(object sender, EventArgs e)
{
    PlayRadio(buttonSW5.Tag.ToString());
}

private void buttonSW6_Click(object sender, EventArgs e)
{
    PlayRadio(buttonSW6.Tag.ToString());
}
```

```

    }

    private void buttonSW7_Click(object sender, EventArgs e)
    {
        PlayRadio(buttonSW7.Tag.ToString());
    }

    private void buttonSW8_Click(object sender, EventArgs e)
    {
        PlayRadio(buttonSW8.Tag.ToString());
    }

    private void buttonSW9_Click(object sender, EventArgs e)
    {
        PlayRadio(buttonSW9.Tag.ToString());
    }

    private void buttonSW10_Click(object sender, EventArgs
e)
    {
        PlayRadio(buttonSW10.Tag.ToString());
    }

    private void buttonSW11_Click(object sender, EventArgs
e)
    {
        PlayRadio(buttonSW11.Tag.ToString());
    }

    private void buttonSW12_Click(object sender, EventArgs
e)
    {
        PlayRadio(buttonSW12.Tag.ToString());
    }

    private void buttonSW13_Click(object sender, EventArgs
e)
    {
        PlayRadio(buttonSW13.Tag.ToString());
    }

    private void buttonSW14_Click(object sender, EventArgs
e)
    {
        PlayRadio(buttonSW14.Tag.ToString());
    }

    private void buttonSW15_Click(object sender, EventArgs
e)
    {
        PlayRadio(buttonSW15.Tag.ToString());
    }

```

```
e)      private void buttonSW16_Click(object sender, EventArgs
        {
            PlayRadio(buttonSW16.Tag.ToString());
        }
    }
```