

Міністерство освіти і науки України
Харківський національний університет радіоелектроніки

Факультет Комп'ютерних наук
(повна назва)

Кафедра Медіасистем та технологій
(повна назва)

КВАЛІФІКАЦІЙНА РОБОТА Пояснювальна записка

рівень вищої освіти перший (бакалаврський)

Розробка UI/UX дизайну інтерфейсу для застосунку транскрибування
(тема)

Виконав:

здобувач 4 року навчання,
групи ВПВПС-21-3



Данііл ЛОЗІН
(власне ім'я, прізвище)

Спеціальність 186 Видавництво та поліграфія
(код і повна назва спеціальності)

Тип програми освітньо-професійна

Освітня програма

Видавничо-поліграфічна справа
(повна назва освітньої програми)

Керівник 
проф. Андрій БІЗЮК
(посада, власне ім'я, прізвище)

Допускається до захисту
Завідувач кафедри МСТ

Жанна ДЕЙНЕКО
(власне ім'я, прізвище)

(підпис)

2025 р.

Харківський національний університет радіоелектроніки

Факультет Комп'ютерних наук
Кафедра Медіасистем та технологій
Рівень вищої освіти перший (бакалаврський)
Спеціальність 186 Видавництво та поліграфія
Тип програми Освітньо-професійна
Освітня програма Видавничо-поліграфічна справа
(шифр і назва)

ЗАТВЕРДЖУЮ:
Зав. кафедри МСТ _____
(підпис)
«19» травня 2025 р.

**ЗАВДАННЯ
НА КВАЛІФІКАЦІЙНУ РОБОТУ**

здобувачеві Лозіну Даніілу Денисовичу
(прізвище, ім'я, по батькові)

1. Тема роботи Розробка UI/UX дизайну інтерфейсу для застосунку транскрибування

Затверджена наказом по університету від 19 травня 2025 р. № 385 Ст


2. Термін подання здобувачем роботи до екзаменаційної комісії 16 червня 2025 р.

3. Вихідні дані до роботи
Вхідний аудіо- або відеопотік, можливо у вигляді файлу; Заявка на розробку інтерфейсу застосунку траскрибування; Вимоги безпеки: шифрування та захист паролем.

4. Перелік питань, що потрібно опрацювати в роботі
Проблематика створення транскрипції, порівняльний огляд існуючих застосунків, сучасні вимоги до UI/UX офлайн-застосунків, постановка цілей та задач; Методики UX-дизайну: Design Thinking, Human-Centered Design, інструменти для створення прототипів, визначення цільової аудиторії та побудова персони користувача, побудова користувацького сценарію взаємодії; Розробка інформаційної структури інтерфейсу, створення схем (Wireframes), створення прототипів (High-Fidelity), обґрунтування стилістичного рішення, розробка інтерактивного прототипу, проведення оцінки користувацького досвіду, внесення змін на основі отриманого зворотного зв'язку; Економічна частина; Висновки.

5. Перелік графічного матеріалу із зазначенням креслеників, схем, плакатів, комп'ютерних ілюстрацій (п. 5 включається до завдання за рішенням випускової кафедри)
Титульний слайд; Актуальність роботи; Аналіз аналогів; Визначення цільової аудиторії; Інструменти для розробки прототипів; Сценарій взаємодії, Інформаційна структура; Схеми (wireframes); Прототипи (High Fidelity); Стилiстичне рішення; Інтерактивний прототип; Тестування з експертами; Внесення змін в дизайн за результатами тестування; Економічна частина; Висновки.


6. Консультанти розділів роботи (п. 6 включається до завдання за наявності консультантів згідно з наказом, зазначеним у п. 1)

Найменування розділу	Консультант (посада, прізвище, ім'я, по батькові)	Позначка консультанта про виконання розділу	
		підпис	дата
Основна частина	проф. Бізюк А.В.		06.06.2025
Економічна частина	ас. Легеза О.М.		27.05.2025

КАЛЕНДАРНИЙ ПЛАН

№	Назва етапів роботи	Термін виконання етапів роботи	Примітка
1	Аналіз завдання на кваліфікаційну роботу, визначення цілей і задач проектування	20.05.2025	виконано
2	Аналітичний огляд у виробництві та застосуванні електронних видань	21.05.2025	виконано
3	Проектування технологічного процесу виготовлення майбутнього інтерфейсу	22.05.2025	виконано
4	Вибір інструментальних засобів розробки	23.05.2025	виконано
5	Проектування інформаційної структури	25.05.2025	виконано
6	Розміщення інформації в інформаційних модулях	27.05.2025	виконано
7	Розробка інтерактивного прототипу інтерфейсу	29.05.2025	виконано
8	Тестування та внесення змін в інтерфейс	30.05.2025	виконано
9	Економічна частина	31.05.2025	виконано
10	Оформлення пояснювальної записки	01.06.2025	виконано
11	Оформлення графічної частини	02.06.2025	виконано

Дата видачі завдання 19 травня 2025 р.

Здобувач 
(підпис)

Керівник роботи  проф. Андрій БІЗЮК
(підпис) (посада, власне ім'я, прізвище)

РЕФЕРАТ

Пояснювальна записка кваліфікаційної роботи: 65 с., 5 табл., 27 рис., 24 джерела.

ТРАНСКРИБУВАННЯ, ЛОКАЛЬНИЙ ОФЛАЙН ЗАСТОСУНОК, UI/UX ДИЗАЙН, ІНТЕРФЕЙС КОРИСТУВАЧА, ДОСВІД КОРИСТУВАЧА, КОНФІДЕНЦІЙНІСТЬ ДАНИХ, ШТУЧНИЙ ІНТЕЛЕКТ, ПРОТОТИПУВАННЯ, ТЕСТУВАННЯ, МЕТОДИКИ ДИЗАЙНУ.

Метою кваліфікаційної роботи є створення інтерфейсу локального офлайн-застосунку для автоматичного транскрибування мовлення з вбудованим штучним інтелектом.

Об'єктом дослідження є процес проєктування зручного й безпечного застосунку для обробки голосової інформації без підключення до інтернету.

Предметом є розробка функціонального та інтуїтивно зрозумілого інтерфейсу, орієнтованого на користувачів із різним рівнем технічної підготовки.

У роботі проаналізовано проблематику розпізнавання мовлення, оглянуто сучасні офлайн-застосунки, вивчено принципи UI/UX дизайну, визначено цільову аудиторію, побудовано схеми wireframes, створено інтерактивний прототип, а також реалізовано функції безпеки. Обґрунтовано стилістичне рішення та проведено тестування з залученням експертів із внесенням подальших покращень спираючись на зворотний зв'язок.

Галузь застосування – автономний програмний продукт для журналістики, освіти, медицини, бізнесу та будь-яких сфер, де обробка чутливого аудіо- та відеоконтенту вимагає високого рівня захисту.

ABSTRACT

Explanatory note of qualification work: 65 p., 5 tab., 27 pic., 24 sources.

TRANSCRIPTION, LOCAL OFFLINE APPLICATION, UI/UX DESIGN, USER INTERFACE, USER EXPERIENCE, DATA PRIVACY, ARTIFICIAL INTELLIGENCE, PROTOTYPING, TESTING, DESIGN METHODOLOGIES.

The aim of the qualification work is to create an interface for a local offline application for automatic speech transcription with built-in artificial intelligence.

The object of the study is the process of designing a convenient and secure application for processing voice information without an internet connection.

The subject is the development of a functional and intuitive interface aimed at users with different levels of technical training.

The work analyses the problems of speech recognition, reviews modern offline applications, studies the principles of UI/UX design, identifies the target audience, builds wireframes, creates an interactive prototype, and implements security features. The stylistic solution is justified and tested with the involvement of experts, with further improvements based on feedback.

The field of application is an autonomous software product for journalism, education, medicine, business, and any areas where the processing of sensitive audio and video content requires a high level of protection.

ЗМІСТ

	С.
ВСТУП.....	7
1 АНАЛІЗ ЗАВДАННЯ НА КВАЛІФІКАЦІЙНУ РОБОТУ	9
1.1 Проблематика створення транскрипцій	9
1.2 Порівняльний огляд існуючих застосунків.....	13
1.3 Сучасні вимоги до UI/UX офлайн-застосунків.....	18
1.4 Постановка цілей та задач.....	21
2 АНАЛІТИЧНИЙ ОГЛЯД ЛІТЕРАТУРИ ЗА ТЕМОЮ	24
2.1 Методики UX-дизайну: Design Thinking, Human-Centered Design	24
2.2 Інструменти для створення прототипів.....	27
2.3 Визначення цільової аудиторії та побудова персони користувача.....	29
2.4 Побудова користувацького сценарію взаємодії	32
3 ОПИС ПРАКТИЧНОЇ ЧАСТИНИ	34
3.1 Розробка інформаційної структури інтерфейсу.....	34
3.2 Створення схем (Wireframes).....	36
3.3 Створення прототипів (High-Fidelity)	39
3.4 Обґрунтування стилістичного рішення.....	46
3.5 Розробка інтерактивного прототипу	49
3.6 Проведення оцінки користувацького досвіду.....	51
3.7 Внесення змін на основі отриманого зворотного зв'язку	53
4 ЕКОНОМІЧНА ЧАСТИНА	56
ВИСНОВКИ	61
ПЕРЕЛІК ДЖЕРЕЛ ПОСИЛАННЯ	63

ВСТУП

У сучасному інформаційному середовищі, де персональні дані, швидкість обробки інформації та автономність стають критично важливими, розробка локальних застосунків із вбудованими ШІ модулями набуває особливого значення. Автоматичне транскрибування аудіозаписів – процес, який ще донедавна вимагав потужних серверів та підключення до хмарних сервісів – сьогодні стає доступним на локальних пристроях, відкриваючи нові можливості для журналістів, викладачів, підприємців та всіх, хто працює з усним контентом.

Однак, попри технічний прогрес, більшість існуючих рішень мають два основні недоліки: вони або вимагають постійного доступу до інтернету, що унеможливорює повний контроль над конфіденційною інформацією, або мають надто складний інтерфейс, орієнтований на досвідчених користувачів. З огляду на це, постає потреба у створенні інтерфейсу, який буде не лише безпечним і функціональним, а й інтуїтивно зрозумілим і зручним навіть для нефахівців в технічних аспектах.

Запропонований застосунок – це дизайн інтерфейсу локальної системи транскрибування з вбудованим штучним інтелектом, який дозволяє не тільки здійснювати точне розпізнавання мовлення, а й підбивати підсумки, аналізувати текст та працювати повністю автономно. Його інтерфейс створено з урахуванням принципів доступного дизайну, що дозволяє задовольнити потреби широкого кола користувачів.

Робота складається з чотирьох основних етапів:

- аналіз проблематики транскрибування мовлення та огляд існуючих офлайн-застосунків, UX-практик та інструментів для створення прототипів;
- визначення цільової аудиторії, побудова користувацьких сценаріїв і створення візуальних схем (Wireframes, UML);

- створення практичної реалізації прототипу інтерфейсу із урахуванням всіх функціональних можливостей, тестування з залученням експертів і адаптація продукту відповідно до отриманого зворотного зв'язку;

- економічне обґрунтування – оцінка витрат на створення дизайну прототипу, аналіз вартості використаного програмного забезпечення та людських ресурсів.

Розроблюваний інтерфейс поєднує в собі мінімалістичний візуальний стиль, інтуїтивну зручність у щоденному використанні та передбачає сучасні механізми захисту даних, включаючи шифрування й локальне зберігання. Такий підхід робить його не просто зручним, а й необхідним інструментом у цифрову епоху, де конфіденційність, безпека та автономність перестають бути опціями – вони стають базовими очікуваннями і стандартами повсякденного життя для кожної людини, незалежно від її технічної підготовки.

1 АНАЛІЗ ЗАВДАННЯ НА КВАЛІФІКАЦІЙНУ РОБОТУ

1.1 Проблематика створення транскрипцій

Автоматичне перетворення усного мовлення на текст набуває широкого застосування у різних сферах – від медичної документації до медіаархівування та освітніх процесів. Однак попри стрімкий розвиток технологій розпізнавання мовлення, на практиці виникають суттєві складнощі. Мовне розмаїття, технічна якість записів, фонові шуми та непередбачуваність живої мови створюють проблеми для точного та швидкого транскрибування.

Саме автоматичне перетворення усного мовлення на текст було назване одним із рекомендованих рішень як профілактичний захід для запобігання негативному впливу виробничих факторів на здоров'я та самопочуття оператора комп'ютерного набору. Це дає змогу знизити навантаження на зір та опорно-руховий апарат працівника, зменшуючи тривале використання клавіатури. Використання транскрибування дозволяє суттєво скоротити час, витрачений оператором на введення інформації, зменшити фізичне навантаження та знизити ризики виникнення професійних захворювань, що підкреслює актуальність розробки ефективних офлайн-рішень у цій сфері [5].

Чимало доступних рішень покладаються на віддалену обробку даних, що передбачає майже постійне підключення до інтернету та передачу конфіденційних записів на зовнішні сервери. Це викликає застереження щодо безпеки, особливо в середовищах, де важливо забезпечити приватність та запобігти витоку інформації. Відповідно, виникає зростаючий запит на інструменти, здатні виконувати транскрибування без підключення до мережі.

Однак доступні офлайн-рішення часто орієнтовані на досвідчених користувачів і мають обмежену зручність у використанні. Це стосується як складності інтерфейсів, так і недостатньої гнучкості в роботі з результатами транскрипції. Для нефахівців, які потребують швидкого й інтуїтивно зрозумілого доступу до функцій, такі рішення залишаються малоприматними.

Ключовим викликом стає не лише якість розпізнавання, а й інтерфейс, який забезпечує зрозумілу навігацію, ефективну взаємодію та стабільну роботу в обмежених умовах. У центрі уваги – потреба у поєднанні технологічної ефективності з практичною доступністю.

Розпізнавання мовлення – це технічне завдання, яке тільки на перший погляд здається тривіальним. У лабораторних умовах, де учасники запису дотримуються чіткої дикції, а звук подається без спотворень, системи транскрипції демонструють вражаючі результати. Проте в реальному середовищі ситуація кардинально змінюється.

Мовлення людей рідко буває стандартизованим. Вимова з різними регіональними чи соціальними акцентами ускладнює завдання системі, яка була натренована на обмеженому наборі мовних зразків. Крім того, жива мова нерідко містить повтори, емоційні вигуки, незавершені фрази, паузи, вставні слова – усе це порушує передбачувану структуру фрази, до якої звик алгоритм. Ще складніша ситуація виникає при багатоголосому мовленні, коли люди перебивають одне одного, або говорять одночасно. В таких умовах навіть людське вухо не завжди однозначно розпізнає зміст, що вже казати про алгоритми[6].

Фоновий шум – ще один потужний фактор, що знижує точність. Навіть незначний шум вентилятора, відлуння приміщення чи вуличні звуки можуть призвести до спотворення ключових елементів мовлення. Онлайн-системи в таких випадках мають перевагу, оскільки можуть використовувати потужні сервери для очищення звуку, нормалізації голосу й компенсації шумів у режимі реального часу. Вони також мають доступ до великих мовних моделей і здатні динамічно оновлювати дані на які спираються при обробці.

Офлайн-застосунки, навпаки, часто обмежені потужністю локального пристрою. Це накладає жорсткі рамки на обчислювальну складність моделей. Тому розробники змушені шукати компроміс між швидкістю, енергоспоживанням і точністю. Через це навіть добре побудована система

розпізнавання на локальному пристрої не завжди здатна відтворити результат, порівнюваний із серверним варіантом.

Усе це підкреслює: ключовий виклик не лише у самій здатності «чути», а в тому, наскільки гнучко система реагує на різноманіття мовленнєвих ситуацій. І хоча нейронні мережі значно просунулись уперед, у реальному світі досі залишаються області, де машинна точність не витримує випробування людською варіативністю.

Залежність від онлайн-сервісів, зокрема в контексті обробки аудіо, створює низку критичних обмежень, які виходять за межі зручності та швидкості. Коли йдеться про персональні або конфіденційні дані, ризики, пов'язані з передачею інформації через мережу інтернет, стають значущими як з юридичної, так і з етичної точки зору.

Особливу увагу в роботі було приділено обмеженням, що виникають унаслідок універсального підходу до відеоматеріалів, які часто поєднують як мовні, так і візуальні сигнали. Це ускладнює ефективну обробку аудіо в ізоляції та підкреслює доцільність розробки вузькоспеціалізованих рішень, зосереджених саме на мовному компоненті[7].

У медичній практиці лікарі можуть мати справу з аудіозаписами, що містять діагнози, симптоматику, персональні дані пацієнтів. Подібна інформація підпадає під захист медичної таємниці, і її витік здатен спричинити як порушення законодавства, так і втрату довіри до системи охорони здоров'я. Те саме стосується юристів, які працюють з клієнтськими запитами, або журналістів, які фіксують розмови зі свідками в умовах, де анонімність є запорукою безпеки.

Передача таких аудіо до хмарного сервісу автоматично створює точку вразливості. Навіть якщо платформа декларує політику конфіденційності, користувач не має повного контролю над зберіганням і обробкою файлів. Технічний витік, людський фактор або навіть запит від урядових структур можуть унеможливити гарантії збереження даних. Для багатьох професій цей варіант не є прийнятним.

Окремо варто зазначити фактор автономності. У регіонах зі слабким покриттям або в умовах відсутності доступу до інтернету (експедиції, військові операції, польова робота) онлайн-обробка просто недоступна. У таких випадках ефективно офлайн-рішення стає єдиним варіантом, здатним забезпечити виконання завдань без технічних компромісів.

Таким чином, попри загальну зручність онлайн-сервісів, у низці практичних сценаріїв саме офлайн-підхід відповідає ключовим вимогам – захисту даних, незалежності від інфраструктури та повного контролю над інформацією. І саме в таких умовах автономні системи транскрибування здобувають не лише перевагу, а критичне значення.

Інтерфейси офлайн-застосунків для транскрибування мовлення часто не враховують потреби користувачів без технічної підготовки. Більшість таких рішень створені з орієнтацією на розробників, дослідників або ентузіастів, які звикли до термінального доступу, специфічної термінології та непрямой взаємодії з програмою. Для користувача, що не має досвіду із командним рядком або не розуміє принципів локального розгортання сервісів, подібні інтерфейси стають суттєвою перешкодою.

Навігація у таких застосунках часто не інтуїтивна, а логіка дій базується на припущенні про знайомство користувача з внутрішніми процесами. Наприклад, від користувача можуть вимагати вручну обрати модель розпізнавання, вказати точний шлях до аудіофайлу або задати параметри через конфігураційний файл. Водночас інші критично важливі опції – як-от формат виводу чи мовні налаштування – можуть бути або прихованими, або реалізованими без зворотного зв'язку.

Таке ускладнення інтерфейсу прямо впливає на поширення офлайн-рішень серед широкого загалу. Викладачі, журналісти, медичні працівники або просто звичайні користувачі, які бажають транскрибувати інтерв'ю, лекції чи особисті записи, не завжди можуть дозволити собі витратити час на технічну підготовку. Вони очікують логіки, подібної до інших звичних застосунків: чітких кнопок, зрозумілих етапів, підказок і зручного перегляду результатів.

Недоступність дружнього інтерфейсу не лише стримує використання технологій, а й фактично обмежує право деяких користувачів на приватне й автономне використання аудіоінструментів. Це створює парадокс: з одного боку, технології досягли значної ефективності, а з іншого – вони залишаються за межами досяжного через недоопрацьовану взаємодію з людиною.

Таким чином, розвиток локальних транскрипційних систем неможливий без переосмислення їхнього інтерфейсу. Зниження порогу входу через спрощення, послідовність і візуальну ясність є як технічним, так і гуманітарним викликом – спрямованим на розширення доступу до технологій як інструменту повсякденної практики.

1.2 Порівняльний огляд існуючих застосунків

З розвитком технологій автоматичного розпізнавання мовлення з'являється дедалі більше інструментів, які дозволяють користувачам здійснювати транскрибування аудіофайлів без необхідності залучати онлайн-сервіси. Однак, кожен з існуючих застосунків має свої особливості, обмеження та переваги, які визначають їх ефективність у різних сценаріях використання.

Для користувачів важливим є не тільки точність розпізнавання та функціональність, але й простота взаємодії з програмним забезпеченням, здатність працювати з обмеженими ресурсами, а також рівень безпеки та конфіденційності даних. Популярні застосунки, такі як MacWhisper, Whisper.cpp і Aiko, мають різний підхід до вирішення цих завдань, що потребує порівняння їхніх можливостей у контексті потреб кінцевих користувачів.

MacWhisper – це застосунок, який здобув популярність завдяки своїй здатності працювати з різними мовами та надавати точні транскрипції навіть у випадках з неідеальним аудіо. Зокрема, він підтримує безліч мов, що дає можливість користувачам працювати з різноманітними аудіофайлами. Застосунок дозволяє здійснювати транскрибування в реальному часі, що особливо зручно для записів інтерв'ю чи лекцій. Однак його точність може знижуватися в умовах шуму чи складного мовлення. MacWhisper обмежений

щодо тривалості аудіофайлів, хоча в більшості випадків ці обмеження не є значними для повсякденного використання. Що стосується підтримки декількох голосів, застосунок здатний розпізнавати кілька голосів, але точність розпізнавання таких записів часто залежить від того, наскільки чітко звучить кожен окремий голос в записі.

Whisper.cpp також пропонує кілька функціональних можливостей для користувачів, зокрема підтримку різних мов і можливість працювати з аудіофайлами різної тривалості. Оскільки цей застосунок орієнтований на обробку аудіо на локальному комп'ютері, він здатний працювати з файлами без постійного підключення до інтернету. Проте він має певні обмеження, зокрема у випадках роботи з аудіо, яке містить сильний фоновий шум або важко сприймані акценти. Для тих, хто потребує точного транскрибування з багатьма мовами або акцентами, Whisper.cpp може бути дещо обмеженим у порівнянні з іншими варіантами. Однак його беззаперечною перевагою є велика гнучкість у роботі з різними форматами аудіо- та відеофайлів, що робить застосунок зручним у використанні для більшості користувачів.

Aiko відзначається своєю здатністю виконувати точне транскрибування навіть в умовах шуму або багатоголосся. Застосунок має хорошу підтримку мов, включаючи як основні світові мови, так і деякі менш розповсюджені. Важливою особливістю Aiko є її адаптація до акцентів, що дозволяє йому точніше розпізнавати мовлення людей з різних регіонів. Він також підтримує транскрибування в реальному часі, що є важливим для використання у потокових трансляціях або живих подіях. Щодо обмежень, Aiko має максимальну тривалість обробки аудіофайлів, що може бути проблемою для деяких користувачів, особливо коли потрібно працювати з довгими записами. Проте підтримка декількох голосів та точність розпізнавання роблять його чудовим вибором для ситуацій, що вимагають високої якості обробки аудіо.

Щодо порівняння інтерфейсу програм, варто звернути увагу на MacWhisper, який вирізняється простотою та орієнтацією на зручність користувача. Він має чітко структуровану головну панель, де основні функції

розташовані у видимих місцях, що дозволяє швидко розпочати роботу. Інтерфейс досить інтуїтивно зрозумілий, навіть для тих користувачів, які не мають глибоких технічних знань. Однак для людей без досвіду у використанні подібних програм може бути певна складність у виборі відповідних налаштувань, особливо коли потрібно змінити мови або спеціалізовані параметри розпізнавання. Загалом, MacWhisper дозволяє швидко завантажити аудіофайл, обрати мову та запустити процес транскрибування без значних складнощів. Проте, для деяких користувачів відсутність налаштувань для глибшої адаптації інтерфейсу може стати обмеженням.

Whisper.cpp, на відміну від MacWhisper, має більш технічний інтерфейс, орієнтований на користувачів з досвідом у роботі з програмним забезпеченням або з певними знаннями в області обробки аудіо. Навігація в Whisper.cpp не є настільки інтуїтивно зрозумілою, і багато налаштувань розташовані в меню, яке може бути незрозумілим для новачків. Для повного використання можливостей застосунку необхідно розуміти, як працювати з командним рядком, що може бути складно для користувачів без технічної підготовки. Хоча Whisper.cpp пропонує велику гнучкість і підтримку різних функцій, його інтерфейс потребує певних знань та навичок для максимально ефективного використання.

Aiko має більш збалансований підхід до дизайну інтерфейсу, пропонуючи простоту для користувачів із різним рівнем підготовки. Головні функції добре виділені і доступні через чітко розташовані кнопки, що дозволяє швидко розпочати процес транскрибування. Однак, незважаючи на зручний загальний вигляд, деякі додаткові функції, такі як налаштування для акцентів або адаптація до різних мов, вимагають додаткових кроків і можуть бути складними для тих, хто не має досвіду в роботі з транскрипційними системами. Інтерфейс Aiko добре підходить для людей, які шукають простоту, але іноді може вимагати трохи більше часу для освоєння тих можливостей, які виходять за межі базового функціоналу.

Кожен з цих застосунків має свої переваги в контексті інтерфейсу. MacWhisper є більш орієнтованим на початківців завдяки інтуїтивно зрозумілим функціям, але може бути обмеженим для користувачів з більш високими вимогами. Whisper.cpp, з іншого боку, є більш гнучким, але його інтерфейс більше підходить для технічно підготовлених користувачів. Aiko є збалансованим рішенням, що підходить для більшості користувачів, але деякі додаткові можливості можуть бути складними для освоєння.

Щодо порівняння цих програм в плані продуктивності, важливо враховувати, наскільки ефективно кожен застосунок використовує ресурси системи та наскільки швидко справляється із завданнями транскрибування. MacWhisper демонструє хорошу продуктивність, ефективно використовуючи ресурси комп'ютера, проте з певними обмеженнями на менш потужних системах. Програма має здатність виконувати транскрибування швидко завдяки оптимізації для використання GPU, що дозволяє зменшити навантаження на CPU. Однак, на слабших пристроях або під час роботи з великими аудіофайлами, можна помітити сповільнення, що впливає на швидкість роботи, особливо при обробці довгих записів. В офлайн-режимі MacWhisper працює стабільно, але на старіших комп'ютерах можуть виникати затримки при обробці складних або багатоголосих аудіо. Адаптація до різних операційних систем є ще одним важливим аспектом: MacWhisper оптимізований більше для macOS, через що на інших платформах може спостерігатись знижена продуктивність, таких як Windows або Linux.

Whisper.cpp, зазвичай, має більшу гнучкість в обробці ресурсів, що дозволяє користувачам налаштовувати параметри використання CPU/GPU згідно з власними потребами. Цей застосунок дозволяє підключати потужніші апаратні ресурси, що може значно прискорити процес транскрибування, особливо при використанні сучасних GPU. Втім, оскільки Whisper.cpp часто потребує роботи з командним рядком та технічних налаштувань, ефективність використання ресурсів може залежати від досвіду користувача. На слабших пристроях або без потужного графічного процесора програмі може не

вистачати ресурсів для оптимальної роботи, що впливає на час обробки аудіо. Що стосується операційних систем, Whisper.cpp забезпечує високу сумісність як на Windows, так і на Linux, однак на macOS може виникнути проблема з деякими драйверами або специфічними налаштуваннями апаратних ресурсів.

Aiko відрізняється своєю здатністю оптимізувати використання апаратних ресурсів при одночасному збереженні стабільності роботи навіть на середньо потужних пристроях. Він вимагає помірного використання GPU, дозволяючи швидко обробляти аудіо в реальному часі без значних затримок. Однак, при роботі з великими файлами чи записами з декількома голосами програма може стикатися з незначними затримками, якщо комп'ютер має низьку продуктивність. Aiko добре адаптований до різних операційних систем, зокрема до macOS, Windows і Linux, що робить його універсальним інструментом для користувачів з різними технічними вимогами.

Таким чином, MacWhisper надає хорошу продуктивність на macOS, але його функціональність зменшується на інших платформах. Whisper.cpp є більш гнучким у використанні ресурсів, однак потребує від користувача певної технічної підготовки для оптимального налаштування. Aiko ж надає оптимальний баланс між продуктивністю та зручністю використання, що робить його доступним для широкого кола користувачів з різними технічними характеристиками комп'ютерів та операційними системами.

Окремої уваги заслуговує аспект конфіденційності, який також впливає на вибір програмного забезпечення, особливо у випадках роботи з чутливими даними. MacWhisper значною мірою орієнтований на забезпечення конфіденційності користувачів, оскільки більшість обробки даних відбувається локально. Це означає, що користувачі можуть виконувати транскрибування без необхідності завантажувати аудіофайли на сервери сторонніх компаній. Такий підхід значно знижує ризики витоку чутливих даних, оскільки всі операції виконуються на самому пристрої. Враховуючи це, користувачі можуть бути впевнені, що їхні аудіозаписи та транскрипції не потраплять до зовнішніх серверів без їхнього відома або погодження.

Whisper.cpp також надає можливість локальної обробки аудіофайлів, що є ключовим фактором для забезпечення конфіденційності. Оскільки програма не вимагає підключення до інтернету для виконання основних функцій, всі аудіофайли та транскрипції обробляються на комп'ютері користувача. Проте, як і у випадку з MacWhisper, безпека також залежить від захисту комп'ютера користувача від зловмисного програмного забезпечення.

Aiko, в свою чергу, забезпечує більш контрольований процес обробки даних з точки зору конфіденційності. Окрім вже згаданого в інших застосунках – локального транскрибування, Aiko має вбудовані механізми для зберігання та шифрування даних, що підвищує рівень захисту.

В усіх трьох програмах – MacWhisper, Whisper.cpp та Aiko локальна обробка даних без передачі аудіофайлів на сервери є основною перевагою, що забезпечує високий рівень конфіденційності та автономності. Функція з додатковим захистом даних застосунка шляхом шифрування точно має входити в перелік функціональних можливостей створеного інтерфейсу.

Важливо зауважити, що у додаткових джерелах, зокрема в одній із тез доповіді, було продовжено розгляд програмного забезпечення для опрацювання голосової інформації, що підтверджує актуальність теми і розмаїття доступних технічних підходів до її реалізації [8].

1.3 Сучасні вимоги до UI/UX офлайн-застосунків

Зручність застосунку безпосередньо впливає на ефективність взаємодії з ним, особливо коли йдеться про офлайн-режим, де ресурси пристроїв зазвичай обмежені та присутня необхідність забезпечення конфіденційності даних. Врахування вимог до інтерфейсу, який поєднує простоту, функціональність та високий рівень безпеки, є важливим кроком на шляху до створення ефективних та зручних рішень для користувачів з різним рівнем технічної підготовки.

У випадку офлайн-застосунків, які обробляють чутливі дані, питання приватності та безпеки мають особливе значення. Користувачі повинні мати

чітке розуміння того, де і як зберігаються їх дані, а також як забезпечено їхню конфіденційність. Інтерфейс застосунку повинен інтуїтивно вказувати місце зберігання даних, чи є можливість їх шифрування або захисту за допомогою пароля. Це дозволяє користувачам відчувати контроль над власною інформацією та забезпечує додаткову безпеку при роботі з чутливими даними.

Крім цього, важливим є надання користувачам можливості керувати своїми даними. Це включає в себе функції для редагування, видалення або переносу інформації. Такі можливості дають змогу користувачам здійснювати контроль над доступом до своїх даних і вирішувати, коли та хто може до них отримати доступ. У випадку офлайн-взаємодії це ще більш важливо, оскільки немає єдиного серверного зберігання, та безпека даних безпосередньо залежить від користувача та можливостей застосунку.

Забезпечення конфіденційності також включає в себе використання надійних методів шифрування, що гарантують захист даних від несанкціонованого доступу. Інтерфейс, який підтримує налаштування шифрування, наприклад як симетричне шифрування AES рівня 256 або ChaCha20 [9], дає користувачам додаткову впевненість у безпеці своїх даних. Водночас, застосунок повинен бути достатньо прозорим, щоб користувачі розуміли, яким чином можливо увімкнути додаткові опції з захисту даних без потреби у складних налаштуваннях або непотрібних кроках.

Також важливо також звернути увагу на загальну зручність використання таких застосунків. Офлайн-застосунки повинні надавати користувачам простий та інтуїтивно зрозумілий інтерфейс, що дозволяє виконувати основні операції без необхідності витратити час на вивчення складних інструкцій або мати технічні навички. У цьому контексті важливо, щоб навігація була логічною та легкою для сприйняття, а користувач міг швидко знаходити потрібні функції.

Одним з аспектів побудови зрозумілих інтерфейсів є мінімалістичний стиль, який не перевантажує користувача зайвою інформацією чи елементами управління. Як сказав відомий промисловий дизайнер у своїй книзі Дітер Рамс:

«Good design is as little design as possible.», маючи на увазі, що хороший дизайн – це не про додавання, а про зменшення [10, с. 116]. Такий підхід дозволяє зосередитися на основних завданнях, що користувач хоче виконати, зменшуючи кількість кроків до мінімуму та забезпечуючи зрозумілість. Важливим є також чітке позначення основних функцій, таких як завантаження файлів, налаштування параметрів або перегляд результатів, що дозволяє користувачеві інтуїтивно розуміти, як працювати з програмою без докладання додаткових зусиль.

Інтерфейс має бути спроектований таким чином, щоб навіть новачки могли швидко освоїтися з усіма функціями. Зрозумілі іконки, добре структуровані меню і наявність підказок чи інструкцій, які не нав'язуються, але доступні, роблять роботу з застосунком більш комфортною. Потрібно прагнути до того, щоб кожен крок був логічним і зрозумілим, без необхідності звертатися до технічної підтримки.

Крім того, інтуїтивність взаємодії забезпечує безперешкодний процес переходу від однієї операції до іншої. Якщо застосунок пропонує користувачеві чіткий, передбачуваний шлях, це значно підвищує задоволення від його використання. Тому, при розробці застосунків, особливо для нефахівців, важливо знайти баланс між функціональністю та простотою використання, що в результаті веде до задоволення потреб користувачів без надмірного навантаження.

UI застосунку має бути доступним для широкого кола користувачів, незалежно від їхнього рівня технічної підготовки. Врахування різних потреб і можливостей користувачів дозволяє створити більш універсальний і зручний інтерфейс. Одним із важливих аспектів доступності є забезпечення підтримки для людей з обмеженими можливостями. Це може включати дотримання певного рівня контрастності за стандартами доступності WCAG (Web Content Accessibility Guidelines) [11] для людей з порушенням зору. Наприклад хорошим тоном буде використання рівня AA який має співвідношення 4.5 : 1

для звичайного тексту та великих елементів, та рівня AAA зі співвідношенням 7 : 1 для менших елементів та текстових підказок.

Продуктивність та швидкість відгуку є критично важливими аспектами UI/UX, особливо для офлайн-застосунків, які працюють в умовах обмежених ресурсів. Користувачі очікують, що застосунок працюватиме швидко та без затримок, навіть при обробці великих обсягів аудіофайлів. Врахування цих вимог означає, що інтерфейс має бути оптимізований для зменшення навантаження на процесор та оперативну пам'ять, щоб не уповільнювати роботу пристрою, на якому працює застосунок.

Один із способів досягнення високої продуктивності – це мінімізація складності інтерфейсу та зменшення кількості зайвих елементів, які потребують додаткових ресурсів для відображення. Простий, функціональний дизайн дозволяє забезпечити швидкий відгук на дії користувача та уникнути зайвих затримок.

Забезпечення високої продуктивності також вимагає від розробників використання оптимізованих алгоритмів та технік, що дозволяють ефективно обробляти інформацію, навіть коли працює велика кількість процесів одночасно. Це особливо важливо при роботі з великими файлами, де затримки або зависання можуть негативно впливати на досвід користувача.

1.4 Постановка цілей та задач

Визначення чітких цілей та завдань на цьому етапі допоможе створити продукт, який відповідатиме вимогам цільової аудиторії та забезпечить ефективну роботу користувача з програмою. Одним із ключових аспектів є визначення основних функцій, а також врахування потреб різних типів користувачів з різним рівнем технічної підготовки.

Спираючись на вищезазначені особливості побудови інтерфейсів та проаналізовані існуючі рішення на ринку можна сформулювати такі цілі:

- створити прототип інтерфейсу застосунку;
- інтерфейс має забезпечувати інтуїтивну навігацію для користувачів;

- інтерфейс має бути виконаний у мінімалістичному стилі, уникати перевантаження інформацією;
- стилістичне рішення має бути втілено у світлих тонах та відповідати сучасним стандартам доступності мультимедіа контенту;
- інтерфейс повинен підтримувати функції безпеки, такі як шифрування даних і захист застосунка паролем

Враховуючи сформульовані цілі роботи, а також актуальні вимоги до проєктування сучасних інтерфейсів, можна виокремити низку конкретних задач, які необхідно реалізувати в межах кваліфікаційної роботи:

- вивчити сучасні методики дизайну;
- визначити цільову аудиторію та створити «персону» користувача;
- продумати сценарії взаємодії користувача, щоб усе було логічно, інтуїтивно зрозуміло і враховувало можливі помилки або непередбачувані ситуації;
- побудувати діаграму для відображення шляхів взаємодії користувача зі застосунком (UML діаграма);
- створити схематичні екрани (wireframes) інтерфейсу із використанням інструментів прототипування;
- створити інтерактивний прототип інтерфейсу з яким можна буде провести тестування;
- реалізувати дизайн для функцій безпеки, включаючи шифрування даних, захист доступу до застосунку паролем, а також функцію повного очищення локальної інформації;
- провести опитування з експертами, щоб отримати зворотній зв'язок;
- внести зміни в прототип, засновуючись на інформації, здобутої під час опитування експертів.

Також важливо врахувати потенційні функціональні можливості розроблюваного інтерфейсу застосунку. Це спрощує процес розробки дизайну та оптимізує час, витрачений на узгодження функцій та налаштувань.

Спираючись на сформульовані цілі та задачі, а також результати аналізу наявних рішень у сфері офлайн-транскрибування, можна окреслити перелік функціональних можливостей майбутнього застосунку. Для зручності реалізації та подальшої навігації ці можливості доцільно згрупувати відповідно до екранів застосунку, на якому вони будуть реалізовані.

Для всіх екранів:

- зручний доступ до кнопки «Назад» або «Закрити»;
- постійний доступ користувача до екранів «Налаштування» й «Історія».

Для екрану обробки файлу:

- можливість скасувати процес транскрибування;
- інтуїтивний прогрес обробки файлу та планований час завершення.

Для екрану результатів та підсумків:

– акцентна кнопка «Підсумувати», другорядна кнопка «Спитати ШІ» та відповідне поле для вводу запитання;

- відображення результату запиту до ШІ та підсумків.

Для екрану налаштувань:

- опція вибору мови застосунку;
- опція вибору мовної моделі для взаємодії з ШІ;
- опція вибору режиму роботи застосунку (швидший/точніший);
- опція вибрати шлях збереження файлів;
- опція увімкнути додаткове шифрування локальних файлів;
- опція увімкнути пароль для входу в застосунок.

Для екрану історії:

- можливість перегляду минулих результатів транскрипцій;
- можливість видалення всіх локальних файлів;
- додаткове підтвердження перед видаленням всіх файлів.

Враховуючи всі ці аспекти, можна створити інтерфейс, який не тільки забезпечить високий рівень зручності та ефективності, але також надасть користувачам необхідні інструменти для захисту своїх даних, що є особливо важливим для тих, хто працює з чутливою інформацією.

2 АНАЛІТИЧНИЙ ОГЛЯД ЛІТЕРАТУРИ ЗА ТЕМОЮ

2.1 Методики UX-дизайну: Design Thinking, Human-Centered Design

Однією з найбільш популярних і ефективних методик є Design Thinking, що фокусується на розумінні потреб користувачів і постійному вдосконаленні продукту через процес ітерації. Поряд стоїть підхід Human-Centered Design, який акцентує увагу на людині як основі кожного процесу розробки, надаючи користувачеві центральну роль у створенні інтерфейсів. Обидві методики сприяють розробці продуктів, які не лише технічно ефективні, але й інтуїтивно зрозумілі та зручні для користувачів. Вони допомагають глибше розібратися в потребах користувачів, створюючи інтерфейси, що мають практичне значення та відповідають високим стандартам доступності та ефективності.

Процес Design Thinking складається з п'яти основних етапів: емпатія, визначення проблеми, ідеяція, прототипування та тестування. Кожен з цих етапів націлений на створення інтерфейсів, орієнтованих на користувача, де важливу роль відіграє розуміння потреб і постійне вдосконалення продукту [2].

Перший етап, емпатія, полягає в тому, щоб глибше зануритись у контекст користувача і зрозуміти його потреби, бажання та проблеми. Це допомагає як дизайнерам так і розробникам створити більш точне уявлення про те, що насправді важливо для кінцевого користувача. Тут важливо не тільки вивчити досвід користувача, а й зрозуміти емоції, які він буде відчувати при взаємодії з продуктом.

Другий етап, визначення проблеми, передбачає чітке формулювання того, що саме потрібно вирішити. Це дозволяє фокусувати увагу на конкретних аспектах продукту, які потребують вдосконалення. Визначення проблеми є важливим етапом, оскільки допомагає уникнути надмірного загального підходу, фокусуючись саме на тому, що реально потребує вирішення.

Ідеація – це етап, на якому UX спеціаліст генерує можливі рішення для визначених проблем. Тут дизайнерський процес фокусується на створенні великої кількості ідей, навіть якщо вони здаються неприродними або складними. Це дає можливість знайти найбільш оригінальні та ефективні підходи до вирішення проблем, відкриваючи нові інноваційні рішення.

Прототипування – це створення швидких макетів, які демонструють, як рішення виглядатимуть у реальному світі. Важливо пам'ятати, що прототипи на цьому етапі не повинні бути ідеальними. Їх мета – перевірити ідеї в реальних умовах, зібрати відгуки та зрозуміти, які аспекти потребують доопрацювання, а які успішно виконують свої функції.

Останній етап – тестування. На цьому етапі прототипи перевіряються на практиці, а зібрані відгуки користувачів використовуються для вдосконалення продукту. Тестування дозволяє виявити недоліки та зрозуміти, чи відповідає продукт потребам користувачів.

Завдяки своїй гнучкості і орієнтованості на користувача, Design Thinking дозволяє створювати інтуїтивно зрозумілі та зручні інтерфейси. Кожен етап процесу вносить свій внесок у розуміння реальних проблем користувачів і дозволяє запропонувати рішення, що підвищують ефективність взаємодії з продуктом. В результаті, цей підхід забезпечує створення продуктів, які реально вирішують проблеми кінцевих користувачів, пропонуючи зручні та прості у використанні рішення.

Human-Centered Design є підходом до проектування, який ставить у центр уваги потреби, досвід та емоції користувачів. Це процес, який враховує не лише технічні аспекти продукту, але й глибокі взаємодії користувача з ним, щоб створити зручний, ефективний і приємний інтерфейс [13].

Принципи HCD базуються на розумінні того, що продукт повинен служити користувачеві, а не навпаки. Першим кроком у цьому підході є дослідження цільової аудиторії, вивчення її потреб, болючих точок та контексту використання. Замість того, щоб фокусуватися лише на технічних можливостях,

HCD допомагає зрозуміти, як користувачі взаємодіють з продуктом, які у них емоції та які труднощі вони зустрічають на кожному етапі взаємодії.

Цей підхід передбачає активне залучення користувачів протягом всього процесу розробки продукту – від початкових етапів дослідження до тестування і фінального вдосконалення. Регулярні зворотні зв'язки від користувачів дозволяють швидко коригувати продукт і робити його більш інтуїтивно зрозумілим, ефективним і доступним.

Інтерфейси, створені з урахуванням принципів HCD, мають високий рівень доступності, адже вони враховують різні рівні навичок користувачів та їхні індивідуальні особливості. Наприклад, застосування ясних вказівок, великих кнопок або контрастних кольорів може значно полегшити взаємодію для людей з обмеженими можливостями. Крім того, дизайн, орієнтований на користувача, знижує ймовірність того, що користувач зіткнеться з труднощами в процесі використання продукту, що покращує його загальний досвід.

Завдяки глибокому розумінню потреб і емоцій користувачів, HCD забезпечує створення продуктів, які вирішують реальні проблеми, а не просто відповідають технічним вимогам. Це дозволяє розробникам створювати інтерфейси, які не лише функціональні, але й приємні для користувачів, що підвищує рівень задоволення та лояльності до продукту. Таким чином, коли людина та її потреби знаходяться в центрі уваги, задача створити ефективний та доступний інтерфейс стає набагато простішою.

Design Thinking і Human-Centered Design включають в себе безліч інструментів і технік, які допомагають створювати продукти, орієнтовані на користувачів. Такими інструментами є наприклад:

- створення персони користувача;
- прототипування;
- тестування та збір зворотнього зв'язку.

Всі ці та інші інструменти працюють разом, дозволяючи дизайнерам отримати глибокі знання про користувачів, їхні потреби і проблеми, а також про способи оптимізації продукту для більш ефективної взаємодії з ним.

У результаті інтеграції обох методик створюються рішення, які не просто «виконують свою роботу», а створюють відчуття, ніби вони вже давно знайомі користувачу. Такі інтерфейси не змушують додумувати, не вимагають навчання, не викликають роздратування – вони підтримують довіру та залучення користувача, перетворюючи технологію на щось по-людськи зрозуміле й близьке.

2.2 Інструменти для створення прототипів

У процесі розробки інтерфейсів важливу роль відіграє вибір відповідного інструменту для створення прототипів. Від цього залежить не лише швидкість реалізації ідей, а й якість взаємодії команди, гнучкість у внесенні змін та зручність тестування. В умовах обмежених ресурсів, обмеженого часу або віддаленої співпраці особливо цінується доступність, інтуїтивність і функціональна насиченість середовища для проєктування. Розуміння ключових можливостей обраного інструменту дозволяє ефективно організувати робочий процес, підвищити продуктивність і зосередитися на створенні зручного досвіду для користувача.

Adobe XD приваблює тісною інтеграцією з екосистемою Adobe, особливо для тих, хто вже працює з Photoshop або Illustrator. Перевагою буде знайомий інтерфейс для користувачів продуктів Adobe, однак обмежена підтримка спільної роботи в реальному часі та припинення його активного розвитку компанією, ставить під сумнів роботу з цим інструментом.

Sketch, орієнтований на macOS, довгий час залишався стандартом у галузі UI-дизайну для «яблучних» систем. Його сила – це простота, велика кількість плагінів і потужних можливостей експорту. Але прив'язка до однієї операційної системи та обмеження щодо спільної роботи створюють бар'єри для гнучких команд.

InVision позиціонує себе як зручний інструмент для інтерактивного прототипування та презентації ідей. За його допомогою легко створювати клікабельні макети, демонстраційні прототипи та сценарії взаємодії, які можна

ділитися з командою чи замовниками для зворотного зв'язку. Платформа підтримує коментування, що полегшує обговорення дизайну без потреби в додаткових засобах комунікації. Проте, у InVision відсутні повноцінні функції редагування графічного дизайну, тому його зазвичай комбінують з іншими інструментами для роботи з візуальною частиною макетів.

Figma є доступним, хмарним сервісом, який підтримує одночасне редагування кількох користувачами в реальному часі. Figma працює у браузері, не вимагаючи встановлення – це знижує технічні бар'єри для входу. Її інтерфейс інтуїтивний, а бібліотеки компонентів і вкрай велика кількість плагінів та шаблонів від суспільства полегшують створення будь-яких інтерфейсів.

У таблиці 2.1 наведено порівняння інструментів для прототипування, та їхні ключові аспекти, які визначають його доречність у тому чи іншому проєкті.

Таблиця 2.1 – Порівняння інструментів для створення прототипів

Критерії	Figma	Adobe XD	Sketch	InVision
Вартість	Безкоштовний план або \$20/місяць	Безкоштовний план або \$34/місяць	\$99/рік	Безкоштовний план або \$199/рік
Платформа	Веб, Windows, macOS	Windows, macOS	macOS	Веб
Спільна робота	Чудова	Обмежена	Через плагіни	Чудова
Прототипування	Чудове	Чудове	Через плагіни	Чудове
Дизайн інтерфейсів	Потужний	Базовий	Потужний	Базовий
Компоненти	Варіативні	Варіативні	Символи	Обмежено
Анімація переходів	Чудова	Добра	Через плагіни	Добра
Плагіни та інтеграції	Велика кількість	Середня кількість	Велика кількість	Середня
Сумісність із розробкою	Висока	Висока	Висока	Середня
Спеціалізація	UI/UX, командна робота	UI/UX, інтеграція з Adobe	Дизайн для macOS/iOS	Презентації, зворотній зв'язок

Наведена таблиця яскраво демонструє переваги Figma, яка пропонує найкращий баланс функціональності, спільної роботи та простоти, що робить її вигідним вибором не тільки у більшості сценаріїв використання UI/UX дизайну, а і в поточному випадку проектування прототипу інтерфейсу.

Особливо беручи до уваги останнє оновлення, презентоване на Figma Config 2025 [14], яке включає декілька інструментів для малювання, взаємодії з векторною графікою і ще більш гнучке створення та налаштування функції автоматичного розташування елементів Auto Layout.

Варто також зауважити, що створення високорівневих мультимедійних продуктів часто потребує використання потужного апаратного забезпечення та додаткових засобів перетворення інформації. Попри те, що в межах цієї роботи основна увага була зосереджена на програмній складовій, ці аспекти залишаються важливими для розуміння загального технічного контексту [15].

2.3 Визначення цільової аудиторії та побудова персони користувача

У процесі створення будь-якого цифрового продукту ключову роль відіграє розуміння тих, для кого він призначений. Без чіткого уявлення про мотивацію, звички, бар'єри та потреби кінцевих користувачів, інтерфейс ризикує бути непрактичним або непридатним для реального застосування. Здатність поставити себе на місце людини, яка взаємодіє з продуктом, вимагає не лише інтуїції, а й системного підходу. Чим точніше окреслено потреби та контекст майбутніх користувачів, тим вищою буде релевантність та ефективність створеного рішення [16].

Побудова інтерфейсу починається не з дизайну, а з глибокого розуміння користувачів, для яких він створюється. Сегментація аудиторії є важливим етапом у створенні інтерфейсу, оскільки кожен користувач має свої унікальні характеристики, потреби та очікування. Одним із основних критеріїв сегментації є вік. Різні вікові групи можуть мати суттєві відмінності в тому, як вони сприймають і використовують технології. Наприклад, молода аудиторія може бути знайома з інтерфейсами на основі мобільних додатків та соціальних

мереж, в той час як старші користувачі можуть надавати перевагу більш традиційним підходам, з чіткими інструкціями та мінімалістичними елементами. У зв'язку з цим, важливо адаптувати інтерфейс, щоб він відповідав уподобанням кожної вікової групи.

Ще одним критерієм є професія користувачів. Професійні потреби та обов'язки можуть впливати на функціональні вимоги до продукту. Наприклад, інтерфейс для програміста або технічного спеціаліста може включати в себе складні функції та детальні налаштування, тоді як для непрофесіонала або користувача, що не має глибоких технічних знань, інтерфейс повинен бути простим і інтуїтивно зрозумілим. Професійна сегментація дозволяє створювати специфічні функції та інструменти для кожного сегмента.

Важливо також визначити чи використовують користувачі продукт для професійних цілей, для розваг чи навчання. Розуміння цих аспектів дозволяє UI/UX дизайнерам створювати більш персоналізовані інтерфейси, що надають користувачам саме те, що їм потрібно для досягнення їхніх цілей. Наприклад, для користувачів, які використовують продукт для роботи, інтерфейс може бути сфокусований на швидкості та ефективності, з простими і зручними інструментами, які дозволяють зосередитись на основних задачах, тоді як для користувачів, що шукають розваги, важливіші елементи візуального дизайну та інтерактивні функції.

Отже, цільова аудиторія застосунка транскрибування така:

- вікове охоплення від 18 до 50+ років;
- за професією: журналісти, студенти, викладачі та підприємці;
- за рівнем технічної підготовки: від початківців до професіоналів;
- моватиція: економія часу, підвищення продуктивності та зручності процесу транскрибування різної кількості матеріалів.

Створення персони користувача також є важливим кроком у процесі розробки продукту, оскільки дозволяє сформулювати чітке уявлення про типових користувачів та їх потреби. Це допомагає тримати фокус на реальних людях, що взаємодіятимуть з продуктом, і сприяє прийняттю рішень, які відповідають

їхнім інтересам. Процес створення персони починається з вивчення демографічних характеристик, таких як вік, місце проживання, які допомагають зробити персонажа більш реалістичним і впізнаваним. Крім того, для створення ефективної персони важливо визначити її болі та бажання. Бажання допомагають зрозуміти, чому користувачі будуть взаємодіяти з продуктом. Це можуть бути прагнення до комфорту, економії часу, поліпшення ефективності або досягнення певної мети. Болі – це труднощі, з якими стикаються користувачі під час вирішення своїх завдань. Усвідомлення болей користувачів дає можливість зосередитися на їх усуненні, щоб створити продукт, який буде корисним і зручним для цільової аудиторії.

Для розроблюваного застосунку, було створено таку персону користувача:



Рисунок 2.1 – Фото персони користувача

Влад, 30 років, проживає в місті Київ. Він є незалежним журналістом та автором статей особистого блогу на платформі Medium. Більшість робочого часу займається організацією та проведенням живих інтерв'ю з політичними діячами, митцями та лідерами думок сучасного суспільства. Після проведення самого інтерв'ю має потребу швидко і зручно конвертувати аудіозапис в текстовий формат.

Його основний біль – коли інформація, або певна її частина з інтерв'ю є конфіденційною, це унеможливорює використання онлайн інструментів, а детально прочитати велике за обсягом інтерв'ю та підбити його підсумки – може зайняти декілька годин.

Основне бажання – мати локальний захищений застосунок для транскрибування зі вбудованою функцією звернення до офлайн-ШІ для підбиття підсумків чи швидкого аналізу транскрибованого тексту.

2.4 Побудова користувацького сценарію взаємодії

Кожен продукт має бути спроектований таким чином, щоб користувачі могли без труднощів досягти своїх цілей, і саме користувацькі сценарії визначають, як саме це відбувається. Процес включає не лише створення логічних шляхів для взаємодії з інтерфейсом, але й забезпечення зворотного зв'язку та визначення точок, де користувачі можуть зустріти труднощі або потребують підтримки. Правильне проектування користувацьких сценаріїв сприяє створенню зручного та ефективного досвіду взаємодії, що дозволяє не тільки задовольнити потреби користувачів, але й забезпечити максимальну продуктивність і зручність у використанні продукту.

Визначення функціональних можливостей є важливою складовою для створення ефективних користувацьких сценаріїв, оскільки вони визначають, що саме повинно бути досягнуто під час взаємодії з продуктом. Для розроблюваного застосунку транскрибування було створено структурну схему для демонстрації його майбутніх функціональних можливостей (рис. 2.2).

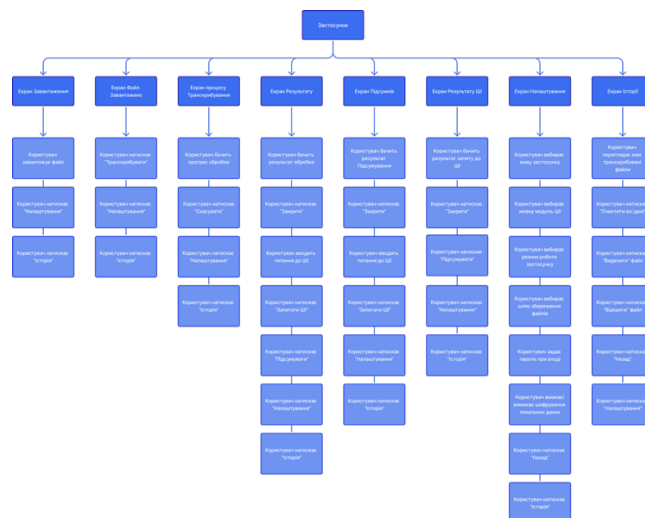


Рисунок 2.2 – Структурна схема

3 ОПИС ПРАКТИЧНОЇ ЧАСТИНИ

3.1 Розробка інформаційної структури інтерфейсу

Поява нових форматів роботи з інформацією, зокрема в аудіо- та відеоформатах, зумовила зростаючу потребу в інструментах, які забезпечують швидке та безпечне перетворення мовлення у текст. Це особливо важливо для журналістів, науковців, правників, дослідників, держслужбовців та всіх, хто працює з інтерв'ю, записами зустрічей, лекціями чи судовими засіданнями. Розробка локального транскрипційного застосунку є обґрунтованим рішенням для тих сценаріїв, де безпека, автономність і контроль над даними мають ключове значення.

На відміну від більшості рішень, що базуються на хмарних сервісах і вимагають постійного з'єднання з мережею, запропонований застосунок повністю функціонує на комп'ютері користувача – без зовнішньої передачі аудіо- або текстових даних. Це дозволяє уникати потенційних ризиків витоку конфіденційної інформації, забезпечити повну автономність та відповідати стандартам інформаційної безпеки в чутливих галузях.

Крім безпеки, важливим є і те, як користувач взаємодіє із програмою. Простий, логічно структурований інтерфейс дозволяє виконати складні технічні операції – транскрипцію, підбивання підсумків, аналітику – з мінімальними зусиллями. У структурі застосунку чітко розмежовано етапи роботи: додавання файлу, запуск транскрипції, перегляд результатів, взаємодія з вбудованим ШІ. Такий модульний підхід забезпечує передбачувану взаємодію, знижує ймовірність помилок та пришвидшує виконання завдань.

Особливу увагу в інтерфейсі приділено не лише основним діям, а й допоміжним інструментам: редагування тексту, можливість ставити питання до транскрибованого матеріалу, створення підсумків, доступ до історії сесій,

реалізація функцій безпеки: шифрування, захист паролем, повне очищення даних – із простим і доступним керуванням для будь-якого рівня користувача.

У процесі проектування користувацького інтерфейсу системи транскрипції ключовим завданням є забезпечення логічної, послідовної та інтуїтивно зрозумілої структури взаємодії. Основний акцент робиться не на візуальному оформленні чи технічних деталях, а на зручності та ефективності роботи з текстовими даними. Кожен етап взаємодії – від завантаження файлу до отримання результатів – має бути організований так, щоб зменшити когнітивне навантаження користувача й забезпечити максимальну концентрацію на змісті та збільшити ефективність застосування.

Початковий етап – екран завантаження файлу виконує роль точки входу в систему. Він має бути мінімалістичним за структурою, водночас містити всі необхідні підказки щодо способу завантаження (включно з функцією drag-and-drop). Запуск процесу транскрипції повинен бути пов'язаний із чіткою та однозначною дією – натисканням кнопки «Транскрибувати», після чого система інформує користувача про зміну статусу, демонструючи орієнтовний час завершення.

Після завершення транскрипції інтерфейс автоматично переходить до наступного логічного етапу – сторінки з результатами. Тут текст відображається у форматі, зручному для читання, з підтримкою копіювання та виділення окремих фрагментів. Особливе значення мають функції контекстної взаємодії, які не вимагають переходу до нових розділів. Зокрема, інструменти «Запитати ШІ» та «Підсумувати» розташовані безпосередньо поруч із текстом, з'являються лише у відповідному контексті (після завершення транскрипції) та виконуються в межах поточного файлу. Така інтеграція сприяє підтримці користувацького фокусу й забезпечує гнучке використання результатів транскрипції без додаткових дій.

Структура інтерфейсу враховує постійний доступ до функції налаштувань та історії. Відповідні іконки розміщені у верхній частині екрана. Це рішення підвищує передбачуваність навігації та формує у користувача

впевненість у послідовності роботи системи. У розділі налаштувань збережено чітку ієрархію [18]: параметри мови застосунку, елементи безпеки (включаючи шифрування локальних файлів та керування паролем), системні функції (вибір мовної моделі, режиму транскрипції, шляхів збереження тощо), усі ці блоки чітко розмежовано для зручного доступу й управління.

Режим роботи системи («Швидкий» або «Підвищена точність») є ще одним параметром адаптивності, який дозволяє користувачеві обрати пріоритет – оперативність або додатковий час для кращої якості розпізнавання. Ця опція не лише підвищує гнучкість системи, а й враховує різні сценарії використання залежно від цілей та ресурсів користувача.

Таким чином, розробка такого застосунку не є простою відповіддю на технічний запит. Це рішення, яке поєднує ефективність, зручність і безпеку в одному продукті. Його доцільність зумовлена як практичними потребами користувачів, так і високими вимогами до захисту інформації.

3.2 Створення схем (Wireframes)

У процесі проектування цифрових інтерфейсів особливе значення має етап візуалізації, коли абстрактні вимоги та функціональні сценарії набувають чітких, структурованих форм. Використання графічних інструментів дозволяє не лише уявити, яким буде вигляд майбутнього застосунку, а й вибудувати логіку взаємодії користувача з системою ще до початку розробки. На цьому етапі формується основа, яка визначає зручність, послідовність і ефективність роботи з інтерфейсом.

Схеми екранів, компоненти навігації, структура даних і розташування функціональних елементів – усе це створюється відповідно до логіки користувацького досвіду. Кожна кнопка, кожен блок або панель виконує роль у загальній композиції, яка має бути не лише естетично привабливою, а й технічно обґрунтованою. Саме тому доцільне використання інструментів, які дають змогу створювати і базові структурні схеми, і деталізовані макети, стає ключовим для якісного результату.

Застосування функції Auto Layout, модульного підходу, повторюваних компонентів та адаптивних розмірів контейнерів сприяє уніфікації дизайну, спрощенню подальшого редагування та підвищенню ефективності командної роботи. Завдяки цьому розробка перестає бути неорганізованою й перетворюється на циклічний процес перевірки й вдосконалення, де кожне рішення може бути швидко протестоване в інтерактивному прототипі.

Процес створення схем інтерфейсу починається з формування базових каркасів – Wireframes, які слугують візуальним орієнтиром для розміщення основних елементів. У Figma цей етап реалізується через створення простих, але логічно структурованих фреймів, що відповідають цільовим роздільним здатностям пристроїв, зокрема 1920×1080 пікселів. Роздільна здатність екрану 1920×1080 пікселів була включена до роботи як найбільш поширена серед користувачів настільних ПК по всьому світу, сягаючи 23.54%, з великим відривом від наступного місця, яке обіймає роздільна здатність 1563×864 пікселів з 10.46% охопленням ринку [19]. Такий підхід одразу орієнтований на екранну реальність, у якій працюватиме майбутній інтерфейс.

Встановлення вертикального напрямку (Vertical) функції Auto Layout підтримує природний порядок дій користувача: спершу подається інформаційний текст основної взаємодії перетягування фалу (Drag&Drop) [20], а нижче кнопка для додавання файлу вручну. Цей напрямок взаємодії відповідає очікуванням користувача та сприяє зменшенню когнітивного навантаження.

Застосування Auto Layout також спрощує масштабування компонентів, оскільки зміни розміру або структури не вимагають ручного вирівнювання елементів. Замість статичних блоків дизайнери оперують гнучкими контейнерами, які реагують на зміну контенту. Це надзвичайно важливо на ранньому етапі, коли інтерфейс зазнає постійних змін.

Схеми Wireframes було створено для послідовного переходу до більш деталізованих прототипів, забезпечуючи при цьому структурну цілісність і передбачуваність поведінки інтерфейсу як зображено на рисунках 3.1 та 3.2.

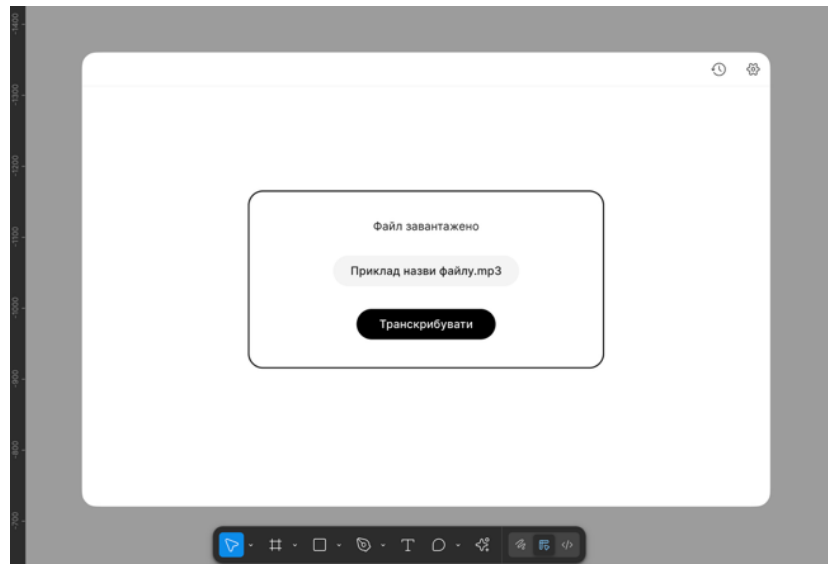


Рисунок 3.1 – Wireframe початкового екрану

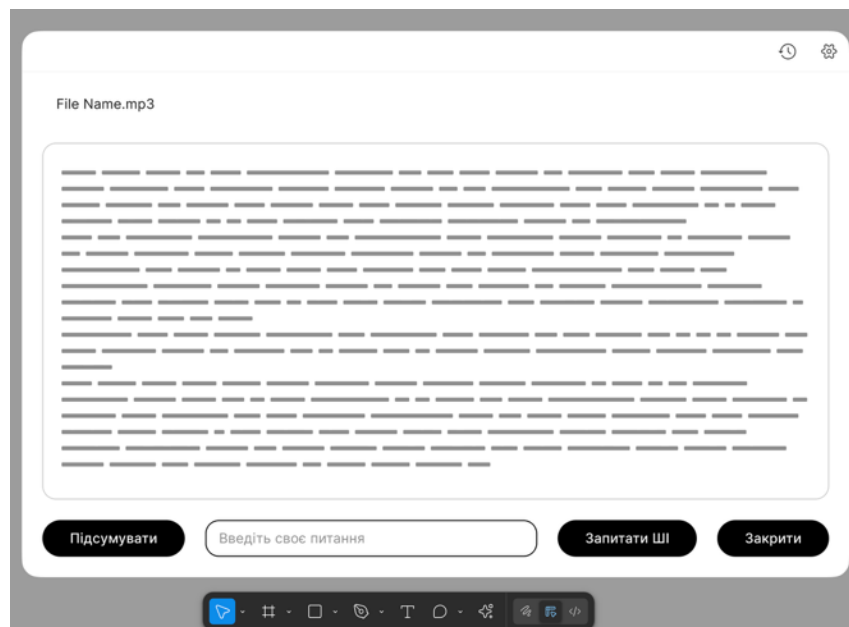


Рисунок 3.2 – Wireframe екрану результату транскрибування

Уніфікація елементів інтерфейсу – ключовий принцип побудови цілісного користувацького досвіду. Для цього у Figma застосовується підхід створення компонентів [21], що дозволяє зберігати єдність стилю, логіки й взаємодії між усіма екранами застосунку на яких застосований цей компонент.

У межах розробки цього інтерфейсу особлива увага приділяється глобальним елементам, таким як меню навігації з іконкою «Налаштування» та «Історія», яке з'являється на кожній сторінці (рис. 3.3).



Рисунок 3.3 – Меню навігації з іконкою «Налаштування»

Меню навігації, яке буде використане повторно, проектується у вигляді компонента з варіантами. Це дає змогу не лише стандартизувати розміщення й стилі, а й полегшити масштабування проекту. Наприклад, при зміні кольору або форми іконки налаштувань, достатньо відредагувати лише головний компонент – і всі екземпляри цієї іконки миттєво оновлюються у всіх місцях використання компоненту.

Іконки завантажуються через плагін SVG Repo – інструмент, який надає доступ до великої бібліотеки графічних елементів з відкритою ліцензією, що дозволяє безкоштовне особисте та комерційне використання. Це не лише економить час, а й дозволяє підтримувати візуальну цілісність без залучення джерел які потребують додаткової покупки ліцензії. Створення окремого компонента для навігаційного меню із цими іконками забезпечує їхню багаторазову інтеграцію в інші екрани інтерфейсу без дублювання роботи.

Централізоване управління компонентами мінімізує ризик візуальних розбіжностей, прискорює редагування й спрощує тестування. В результаті створюється послідовний, зручний та передбачуваний інтерфейс, який легко адаптується до змін, зберігаючи при цьому високу якість отриманого досвіду.

3.3 Створення прототипів (High-Fidelity)

Після того як загальна структура інтерфейсу була визначена через Wireframes, наступним кроком є створення високодеталізованих прототипів (High Fidelity Prototypes) [22], що враховують типографіку, стилі кнопок, палітру кольорів і інші візуальні елементи. Це дає змогу не лише перевірити вигляд інтерфейсу, а й оцінити його функціональність на більш високому рівні деталізації.

Одним із важливих аспектів є створення кнопок, таких як «Вибрати файл», які стають важливими точками взаємодії користувача з додатком. Для цієї кнопки, використовується Auto Layout, що дозволяє створити гнучкий та адаптивний елемент. Текст основної взаємодії, який є центрованим, оформлено темно-сірим кольором з дотриманням контрасту. Блок з кнопкою містить текстовий елемент, з відступами 32px та 12px по горизонталі та вертикалі відповідно, а також округленими кутами на 24px. Ці властивості додають кнопці чіткість, зручність у використанні та візуальну елегантність, кнопку також центровано, що підтримує логічну ієрархію елементів інтерфейсу, як показано на рисунку 3.4.

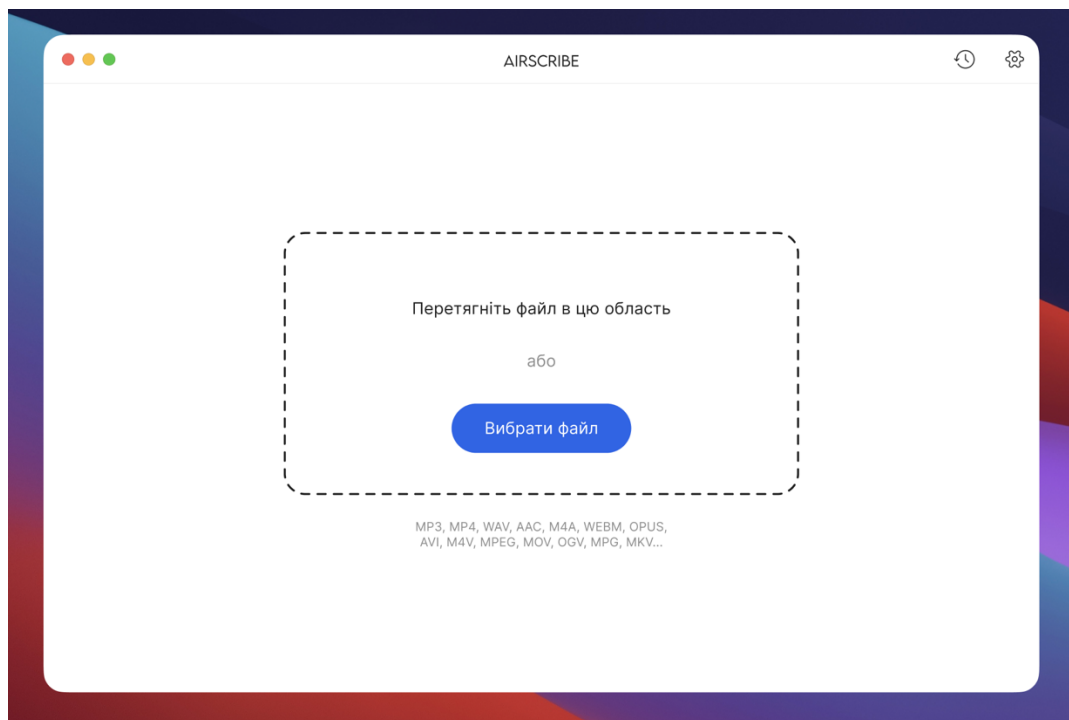


Рисунок 3.4 – Дизайн початкової сторінки з кнопкою «Вибрати файл»

На екрані завантаження файлу, після успішного процесу завантаження показується компактний та інтуїтивно зрозумілий блок підтвердження з іменем доданого файлу (Приклад назви файлу.mp3), що оформлено у вигляді сірого блоку з текстом. Під ним – інформативний текст «Файл завантажено», який має сірий колір та не перевантажує зір. Основна дія – синя кнопка «Транскрибувати», яка привертає увагу і чітко показує наступний крок.

Всі елементи цього екрану обрамлені пунктирною рамкою з округленими кутами, що створює ефект активної зони взаємодії з головного екрану, але вже з меншим контрастом для привернення меншої уваги (рис. 3.5)

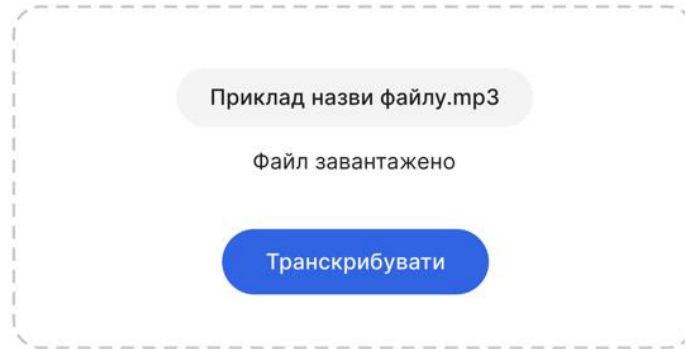


Рисунок 3.5 – Дизайн сторінки з доданим файлом

На сторінці транскрибування додається візуальний індикатор прогресу у вигляді лінії товщиною 8px зі скругленими кінцями, що чітко відображає поточний стан обробки файлу (рис. 3.6). Це не тільки інформує користувача про етап виконання операції, але й гармонійно вписується в загальний дизайн, надаючи йому динамічності.

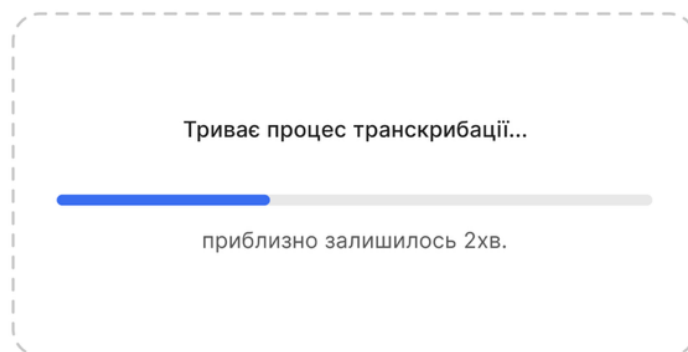


Рисунок 3.6 – Дизайн сторінки транскрибування

На екрані результату транскрипції реалізовано чистий і зосереджений блок із текстом, який легко читається завдяки шрифту Inter, який забезпечує хорошу щільність та сучасний вигляд (рис. 3.7). Основна частина – це великий

текстовий блок, що виводить розпізнаний текст з аудіо. Він не має зайвих візуальних відволікань, завдяки чому фокус користувача тримається на змісті.

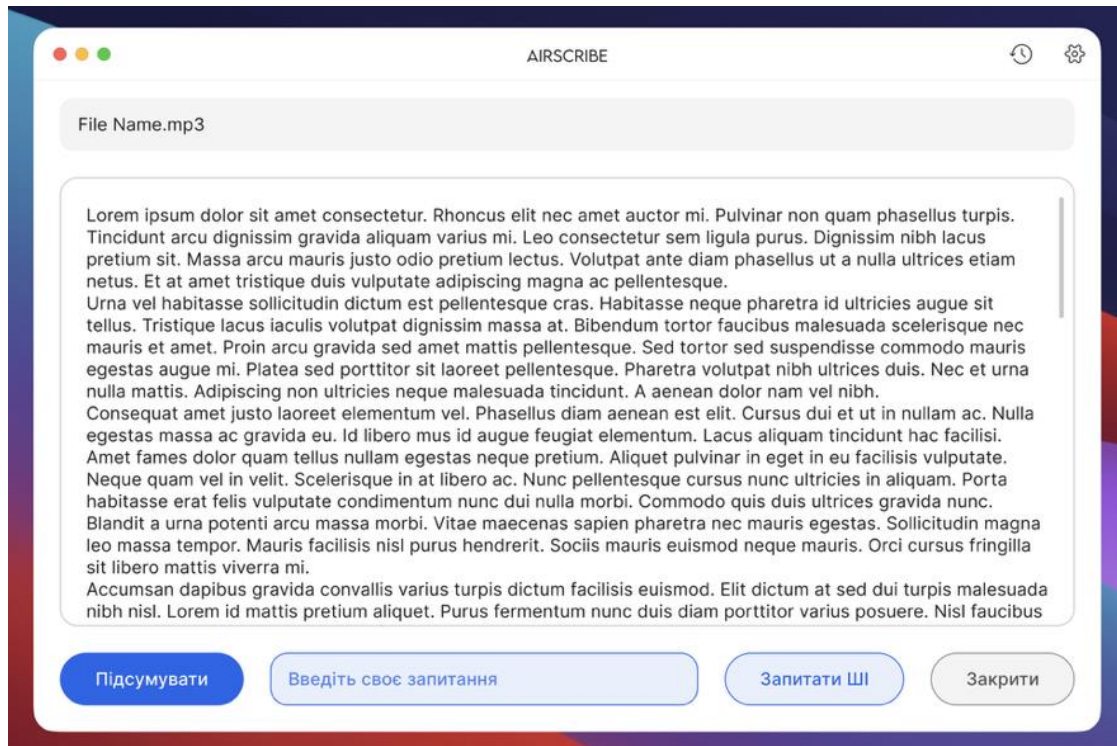


Рисунок 3.7 – Сторінка результату транскрибування

Нижче – два основні елементи взаємодії.

1. Синя кнопка «Підсумувати» – візуально акцентована як основна дія, що дозволяє автоматично згенерувати стислий виклад розмови.

2. Елемент взаємодії у вигляді поля для введення із заокругленими кутами, м'якого синього кольору. Він містить текст «Введіть своє запитання», який спонукає користувача взаємодіяти. Праворуч від поля розміщено синю кнопку «Запитати ШІ». З правого краю присутня сіра кнопка «Закрити» – менш виразна, але інтуїтивно зрозуміла як другорядна дія, по натисканню на яку екран перегляду результату буде закритий, а інтерфейс перейде до початкового екрану та буде очікувати на завантаження файлу.

Увесь текстовий блок логічно згрупований, вирівняний по центру та візуально збалансований. Його та елементи взаємодії обрамлено відступом в 24px від інших елементів для відчуття завершеності та зони взаємодії (рис. 3.8).

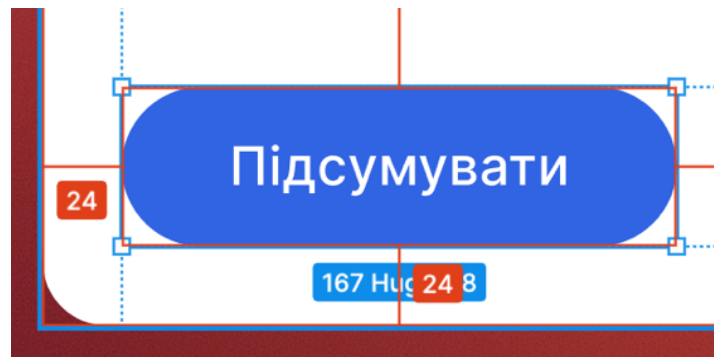


Рисунок 3.8 – Відступ блоку елементів у 24px

З цієї ж сторінки користувач може отримати доступ до створення підсумків та написання запиту до ШІ, що дозволяє швидко поставити запитання на основі транскрипції – наприклад, для уточнення інформації або детальнішого аналізу. Обидві функції відіграють ключову роль в роботі застосунку транскрибування, реалізацію дизайну цих екранів можна побачити на рисунках 3.9 та 3.10.

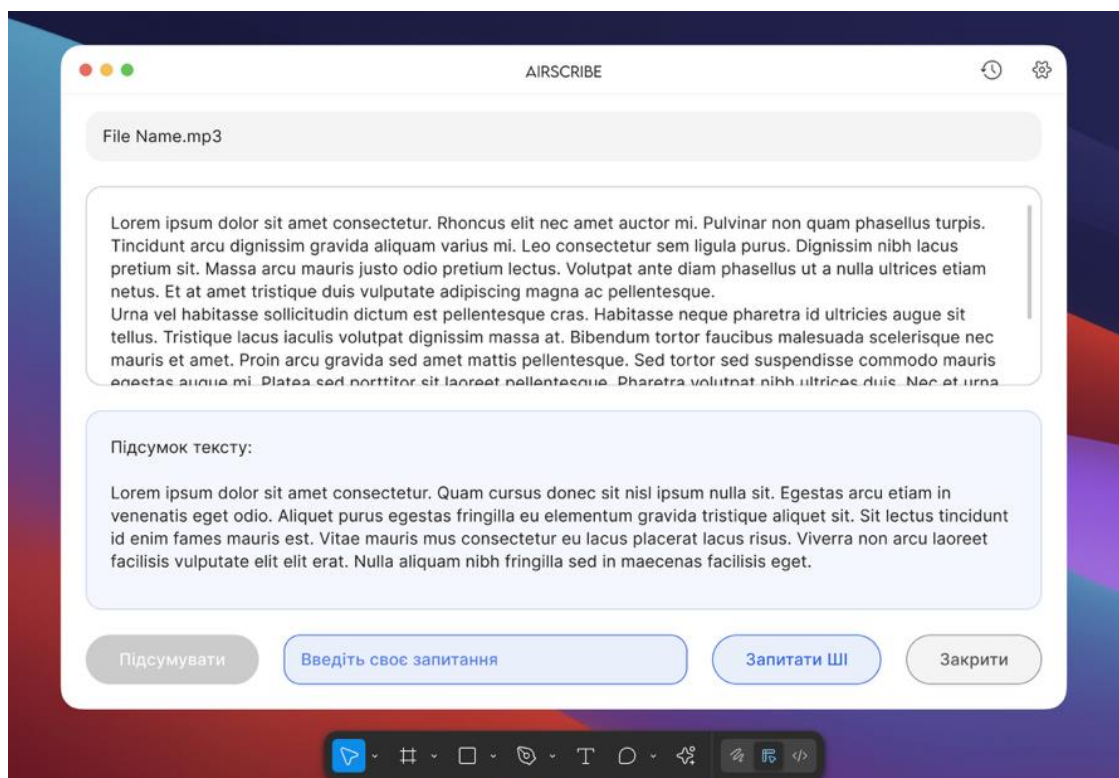


Рисунок 3.9 – Сторінка «Підсумки»

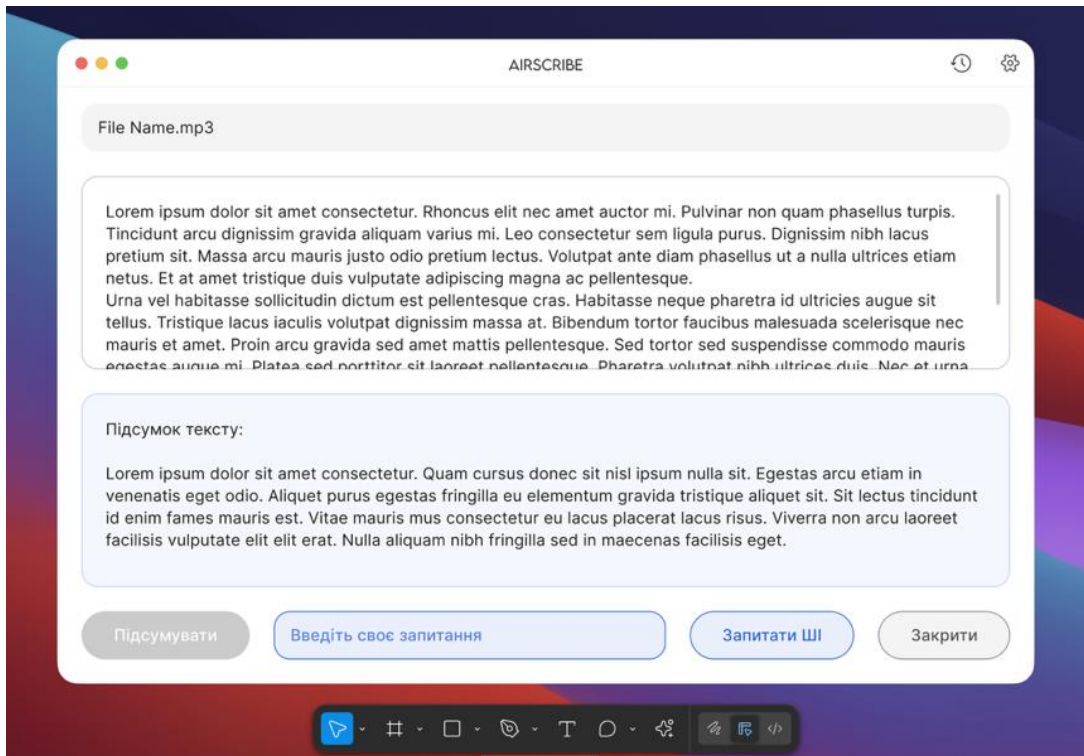


Рисунок 3.10 – Сторінка «Результат ШІ»

На екрані налаштувань реалізовано багатофункціональний інтерфейс з інтуїтивною структурою елементів (рис. 3.11).

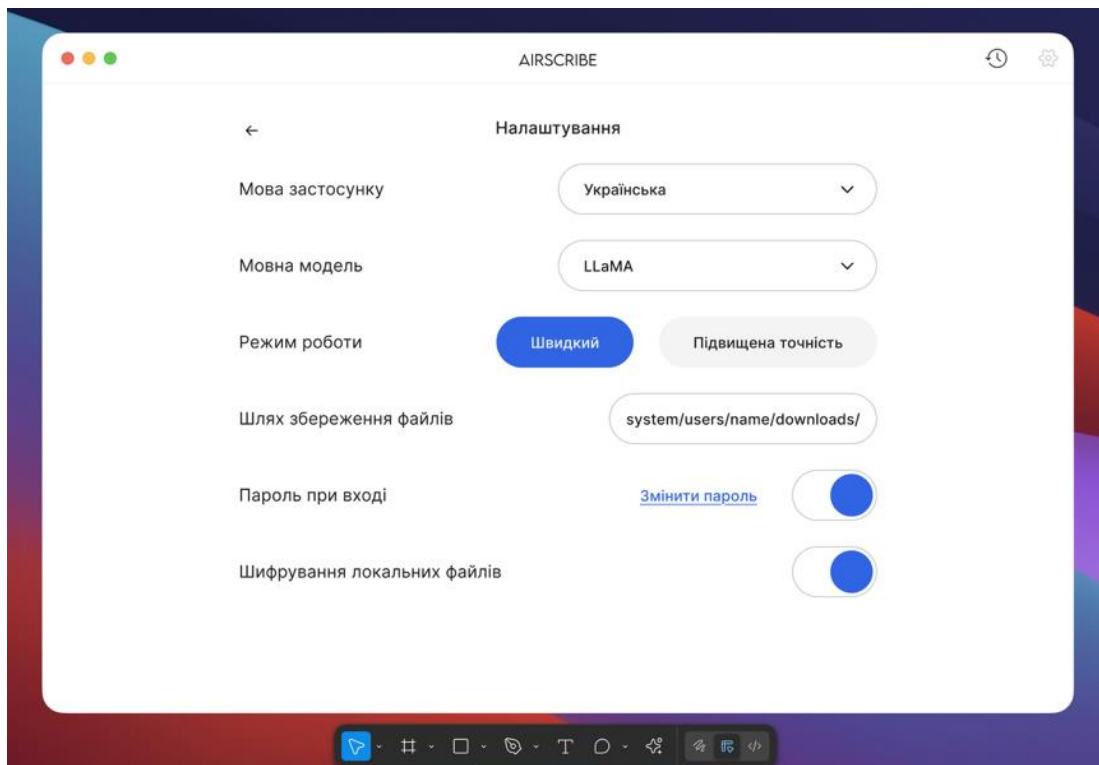


Рисунок 3.11 – Сторінка «Налаштування»

Вгорі – великі поля вибору мови застосунку та локальної мовної моделі у вигляді випадаючих списків. Під ними – перемикач для вибору режиму роботи «Швидкий» або «Підвищена точність», який стилізований під основний акцентний колір застосунку. За цим розташоване поле для задання шляху збереження файлів. Кожен елемент має велику кнопку або область взаємодії, що значно зменшує відсоток похибки користувача. Інтерактивні перемикачі для активації паролю при запуску та шифрування локальної бази мають сучасний вигляд і плавну зміну стану.

На екрані історії відображено організований список вже транскрибованих файлів. На кожному файлі присутня опція «Видалити» та «Відкрити» для повторного перегляду. Вгорі реалізовано кнопку з іконкою «Назад» для повернення на екран на якому користувач перебував перед відкриттям історії. Внизу присутня кнопка «Видалити всі файли» яка має відтворювати відповідну функцію – за потреби користувача стирати всі локальні файли створених транскрипцій. Дизайн екрану історії відображений на рисунку 3.12.

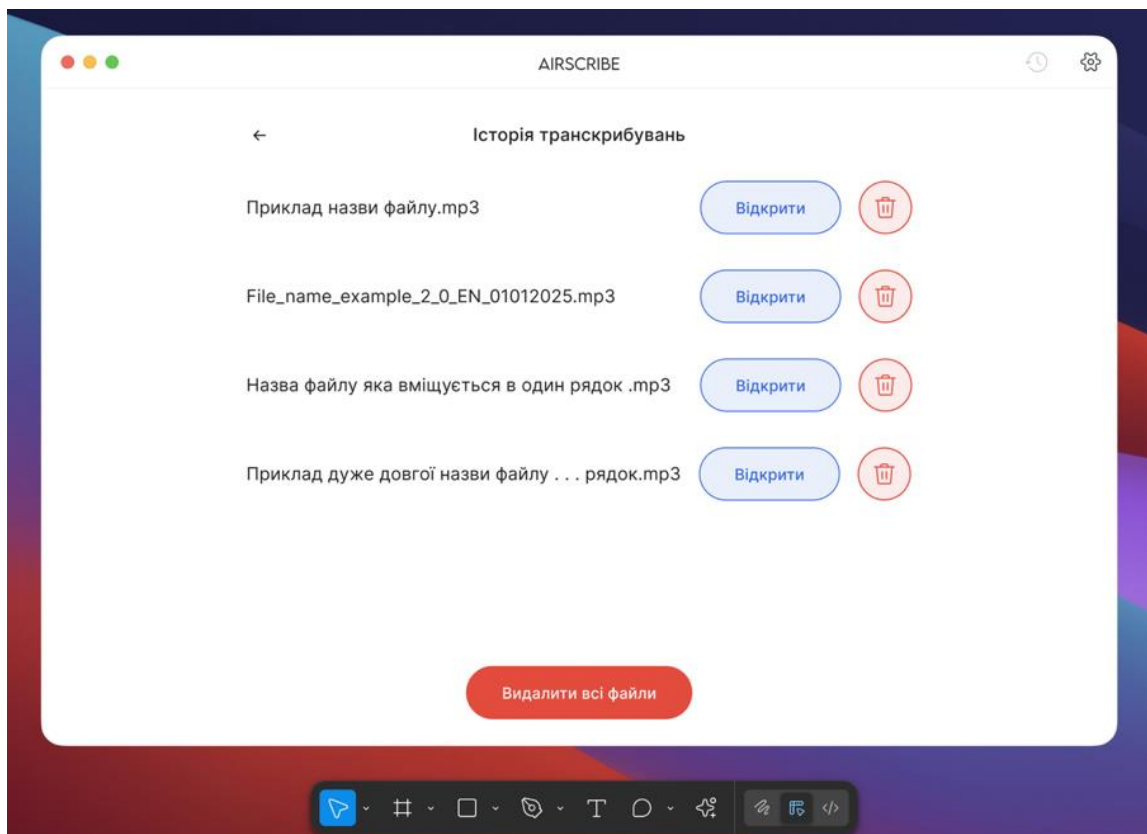


Рисунок 3.12 – Сторінка «Історія»

Очевидно що така дія, як видалення всіх локальних файлів не може відбутись за один клік. Саме тому для перешкоди випадковому натисканню на цю кнопку було впроваджено дизайн додаткового модального вікна з підтвердженням цієї дії (рис. 3.13). Модальне вікно чітко описує що станеться при підтвердженні цієї дії, що запобігає отриманню неочікуваного результату у випадку небажаного натискання. Відповідно з користувачем іде діалог, в якому зрозуміло, що дію буде неможливо скасувати, основна кнопка «Видалити» має червоний колір саме з причини її важливості на взаємодію з інтерфейсом. Проте, якщо користувач просто помилився, то присутня зручна другорядна кнопка «Назад», яка, як і очікується, закриває це діалогове модальне вікно на повертає користувача на звичайний вигляд екрану «Історія».

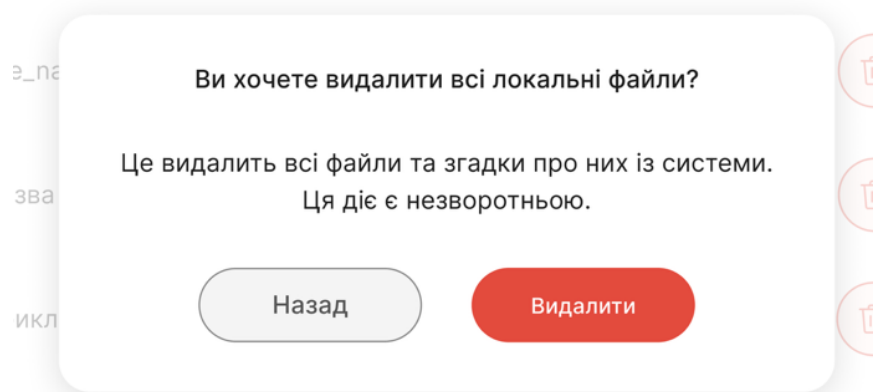


Рисунок 3.13 – Модальне діалогове вікно

3.4 Обґрунтування стилістичного рішення

Візуальне оформлення інтерфейсу відіграє ключову роль у сприйнятті продукту, впливаючи не лише на естетику, а й на зручність взаємодії. Правильно підібрані кольори та структуровані компоненти формують відчуття цілісності, допомагають користувачеві орієнтуватися та швидко приймати рішення. Кожен елемент, від відтінку кнопки до стилю підказки, несе функціональне навантаження, створюючи логічну ієрархію та підсилюючи якісний користувацький досвід.

Вибір кольорової палітри для інтерфейсу застосунку транскрибування ґрунтується на принципі функціонального маркування елементів та забезпечення візуальної ієрархії. Кожен колір не лише виконує естетичну функцію, а й спрямований на формування очікувань користувача щодо характеру дії чи статусу елемента. Вибір кольорів відбувався з урахуванням їхньої психологічної інтерпретації та здатності підтримувати доступність інтерфейсу для максимальної зручності користувачів.

Сірий та темно-сірий колір призначений для нейтральних, не важливих або другорядних дій (рис. 3.15). Його застосування спостерігається в елементах, що не потребують негайної уваги або не передбачають змін у системі: пояснювальні написи, кнопка «Закрити». Цей вибір знижує візуальне навантаження та дозволяє сфокусуватися на більш активних діях.



Приклад назви файлу.mp3

Файл завантажено

Рисунок 3.15 – Застосування сірого кольору

Синій колір є акцентним та саме він виконує роль позначення активної дій або опцій, пов'язаних із прямою взаємодією з застосунком (рис. 3.16). Його використано для оформлення «Call to Action» кнопок «Транскрибувати», «Підсумувати», шкали прогресу транскрибування, а також у вигляді поточного вибору в виборі мовної моделі та стану перемикачів на екрані «Налаштування». З психологічної точки зору, цей колір традиційно асоціюється з надійністю, спокоєм і безпечністю [23].



Транскрибувати

Рисунок 3.16 – Застосування синього кольору

Червоний колір використано для критичних або незворотних дій, що потребують усвідомленості з боку користувача (рис. 3.17). Зокрема, це кнопка «Скасувати» під час транскрибування, а також дія «Видалити всі файли» на екрані історії файлів. Завдяки високій емоційній інтенсивності, червоний відразу привертає увагу, попереджаючи про важливість рішення.



Рисунок 3.17 – Застосування червоного кольору

Типографіка, як складова частина стилістичного оформлення, виконує роль структурного каркасу інтерфейсу. Поєднання декоративного шрифту LEMON MILK із класичним Inter дозволяє водночас надати інтерфейсу індивідуальності та зберегти читабельність і функціональність. LEMON MILK чудово підходить для брендovаних елементів, як логотип (рис. 3.18), тоді як Inter забезпечує чітку ієрархію в текстових елементах по всьому застосунку.

AIRSCRIBE

Рисунок 3.18 – Шрифтовий логотип

Заголовки вирізняються підвищеною товщиною шрифту, що дозволяє швидко ідентифікувати ключові блоки. Допоміжні текстові елементи мають сірий колір і стандартну вагу, підтримуючи візуальну ненав'язливість та слугуючи фоном для основних елементів. Інтерактивні посилання, як-от «Змінити пароль», оформлюються синім кольором і підкресленням, що відділяє їх від іншого тексту та позначає як заклик до дії.

Таким чином, кольори та шрифти в інтерфейсі не існують ізольовано, а взаємодіють як система, що підвищує ефективність навігації, скорочує час на прийняття рішень і забезпечує візуальну узгодженість усіх екранів застосунку.

Модульність та послідовність в елементах застосунку, була досягнута в більшості завдяки функції інструменту Figma – Auto Layout, вона особливо помітна в секціях з багатьма різними елементами, які потребують логічного компоновання. Наприклад, у блоці налаштування (рис 3.19). Це дозволяє візуально об'єднати функціонально пов'язані елементи, підтримуючи послідовність взаємодії користувача. Таке структурування полегшує зміну розмірів і розташування елементів залежно від пристрою або контексту використання, зберігаючи цілісність інтерфейсу.

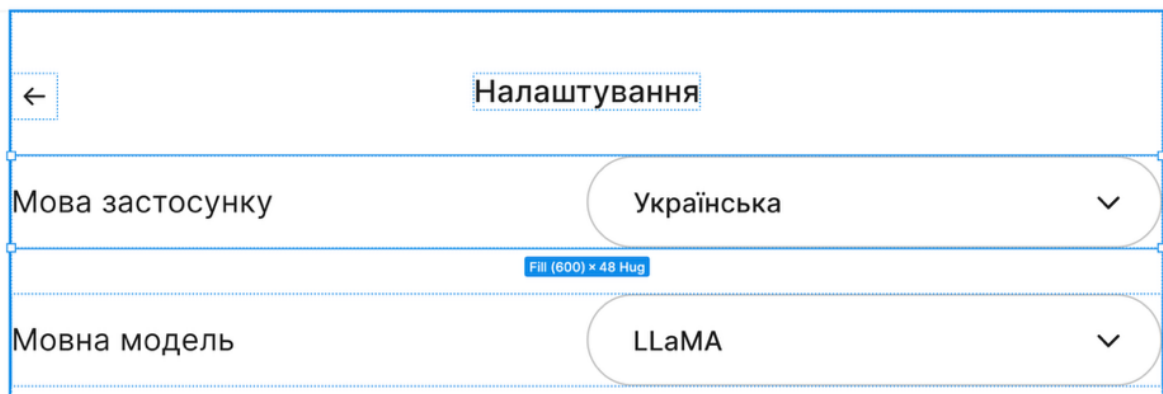


Рисунок 3.19 – Застосування Auto Layout на екрані «Налаштування»

3.5 Розробка інтерактивного прототипу

Після завершення розробки візуальної частини макетів основних екранів наступним важливим етапом є створення інтерактивного (клікабельного) прототипу, який дозволяє перевірити UX-логіку та взаємодію користувача з інтерфейсом. У Figma це здійснюється через вкладку Prototype [24], де встановлюються інтерактивні переходи між сторінками. Така інтерактивність дозволяє створити реалістичну модель навігації між екранами і дає змогу користувачеві симулювати процес взаємодії зі застосунком (рис. 3.10).

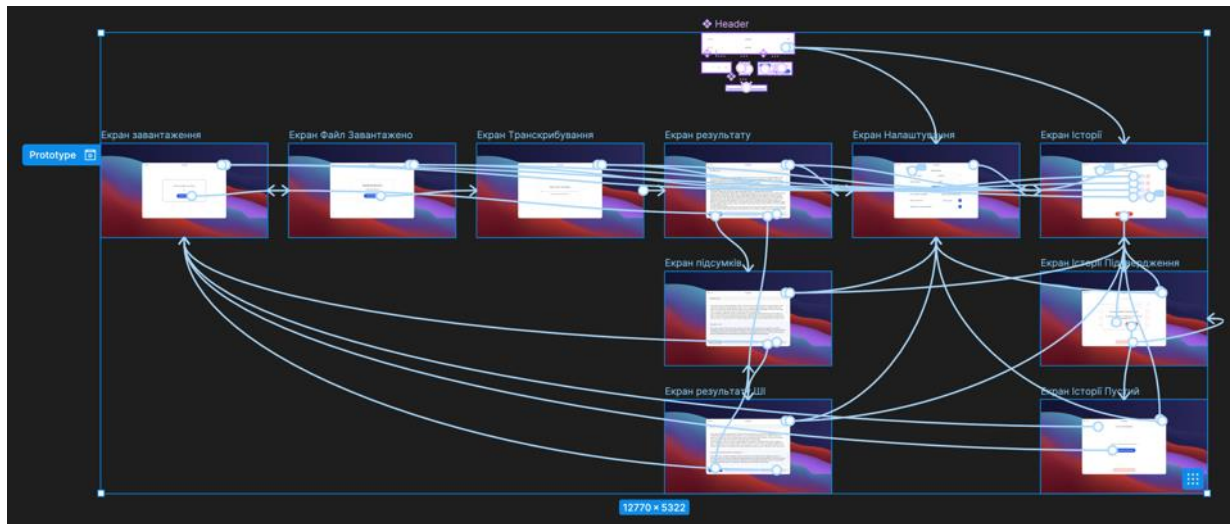


Рисунок 3.14 – UX-логіка прототипу у Figma

Прототип дає можливість перевірити послідовність кроків, яку проходить користувач. Наприклад, він починає з додавання файлу, спостерігає за процесом транскрибування, а потім переходить до сторінки з результатами, де може задати питання або зробити підсумок. Завдяки інтерактивності, користувач може взаємодіяти з усіма основними елементами інтерфейсу, таким чином перевіряючи, чи відповідає навігація очікуванням і чи інтуїтивно зрозумілий сам процес.

Також важливим елементом є доступ до налаштувань та історії, їх було реалізовано на кожному екрані за допомогою компоненту, який посилається на ці екрани. Це дозволяє користувачеві завжди бути в курсі можливостей для налаштування застосунку, незалежно від того, на якому етапі перебуває процес транскрибування чи користувач переглядає результати.

Інтерактивне прототипування дає змогу тестувати не лише візуальні складові інтерфейсу, а й перевіряти саму UX-логіку – наскільки зручно користувачу орієнтуватися у додатку, чи логічно розташовані функціональні елементи і чи немає зайвих кроків чи кліків. Цей етап дозволяє мінімізувати потенційні проблеми з навігацією та зручністю використання ще до початку реалізації на етапі розробки. Тестування прототипу дає змогу виявити будь-які недоліки та уточнити деталі, що можуть призвести до непорозумінь або труднощів у процесі взаємодії з кінцевим продуктом.

3.6 Проведення оцінки користувацького досвіду

У межах розробки дизайну застосунку було проведено опитування серед експертів з потенційної цільової аудиторії, яка включає: UX експертів, підприємців, журналістів та викладачів, з метою отримання зворотного зв'язку щодо взаємодії з інтерфейсом. Щоб з'ясувати, наскільки обраний дизайн відповідає потребам потенційних користувачів, було залучено п'ятьох експертів. Усі вони мали практичний досвід роботи зі вже існуючими інтерфейсами транскрибування, що дало змогу зібрати коментарі не лише з точки зору естетики, а й технічної доцільності розробленого прототипу.

У всіх опитуваних було отримано згоду на відображення їхніх відгуків не в анонізованому вигляді для представлення в кваліфікаційній роботі. Залучених до опитування експертів зазначено в таблиці 3.1.

Таблиця 3.1 – Залучені до опитування експерти

Прізвище та ім'я	Посада	Напрямок використання застосунку
Поляк Євгеній	SEO та співзасновник української компанії AESTAR	Транскрибування довгих зустрічей з клієнтами, більшість з яких є фінансовими та конфіденційними
Бойко Тарас	UX керівник команди та експерт з контролю якості в компанії AESTAR	Доступ до можливості транскрибування як з ПК так і з ноутбука, детальний аналіз зустрічей, підсумування для постановки задач команді розробників
Галькевич Марина	Підприємець у сфері онлайн-ритейлу та логістики	Потреба в простому застосунку для транскрибування онлайн дзвінків, щодо логістичних пересувань товарів
Ягольницький Олександр	К. е. н., доцент. Доцент кафедри економіки та права	Транскрибування та аналіз матеріалів для подальшого викладання, написання лекцій та створення презентації
Перекрыстов Єгор	Журналіст у Dnevnik, словацькій газеті, що знаходиться в Люблянці	Потреба в застосунку, який працюватиме на ноутбуці, для швидкого транскрибування записаних матеріалів інтерв'ю з респондентами

Кожному з учасників було поставлено такі питання:

– «Як ви оцінюєте візуальну привабливість (кольорова палітра, шрифти), за шкалою від 1 до 10, де 1 – це візуально непривабливо, а 10 – це максимально привабливо?»;

– «Чи всі функції застосунку працювали так, як ви очікували? Якщо ні, то опишіть який саме елемент не був інтуїтивно зрозумілим?»;

– «Чи є у вас практичні ідеї як можна покращити застосунок або взаємодію з ним? Якщо так, опишіть більш детально що саме слід покращити».

Відповіді на запитання були отримані та записані в таблицю 3.2.

Таблиця 3.2 – Відповіді на питання

Експерт	Візуальна оцінка	Відповідь на запитання 2	Відповідь на запитання 3
Євгеній Поляк	9	Так, все інтуїтивно. Гарне відображення системних налаштувань	Розділити текст на абзаци, де це доречно, та зробити більші відступи, бо виглядає добре, а читати складно
Тарас Бойко	8	Функції працювали, сподобався візуальний прогрес бар статусу обробки	Текст в полі для запитання III має контраст 4.35:1, він близький до рекомендованого 4.5:1, але таким не є, потрібно виправити цей момент
Марина Галькевич	10	Усі працювали	Не одразу зрозуміла де шлях збереження файлів. Не думаю що це є критичною проблемою
Олександр Ягольницький	9	Так. Особливо сподобалась функція підсумку	Було б добре мати копіювання результатів як транскрипції, так і підсумків
Єгор Перекръостов	10	Так, усе просто і зрозуміло, навіть без інструкцій!	Іноді промахувався по кнопці «Закрити» на екрані результату, було б чудово її перенести в інше місце

Аналіз отриманих відповідей дав змогу сформувати низку конструктивних рекомендацій, частину з яких було швидко реалізовано за допомогою методики ітерації прототипів.

3.7 Внесення змін на основі отриманого зворотного зв'язку

Майже всі опитані позитивно оцінили візуальне оформлення, кольорова палітра була визнана приємною, а обраний шрифт чітким і зручним для читання. Проте були і зауваження щодо покращення.

Більшість ідей та пропозицій покращення стосуються екрану «Результат».

Спираючись на зворотний зв'язок було перенесено кнопку «Закрити» з правого нижнього кута, де була можливість натиснути на неї випадково, у верхній кут, ближче до елемента що відображає назву файлу (рис 3.20).

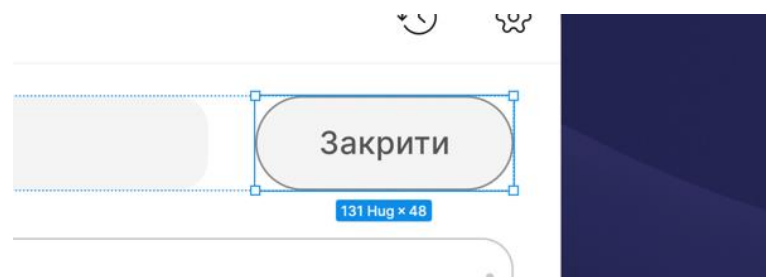


Рисунок 3.20 – Нове положення кнопки «Закрити»

Було виправлено контраст тексту та фону в блоці звернення до ШІ «Введіть своє запитання» з $4.35 : 1$ на більш доступний $5.16 : 1$. Відображення результату можна побачити на рисунку 3.21.

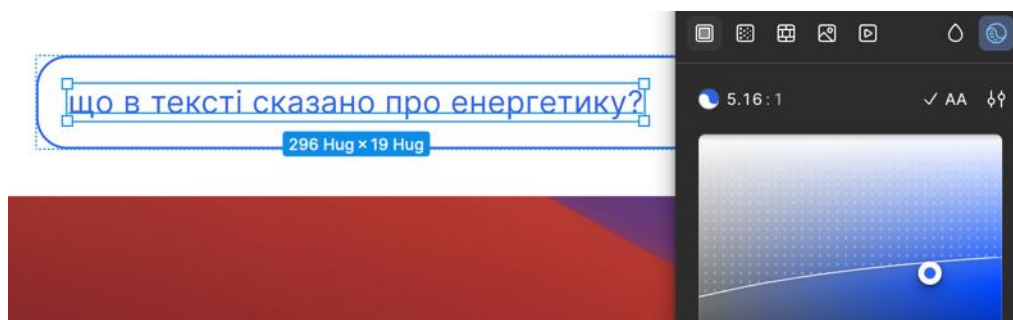


Рисунок 3.21 – Виправлений контраст тексту в блоці звернення до ШІ

Також було змінено структуру відображення результатів транскрипції: замість суцільного блоку тексту запроваджено поділ на логічні абзаци з чіткими візуальними відступами червоної строки (рис. 3.22).

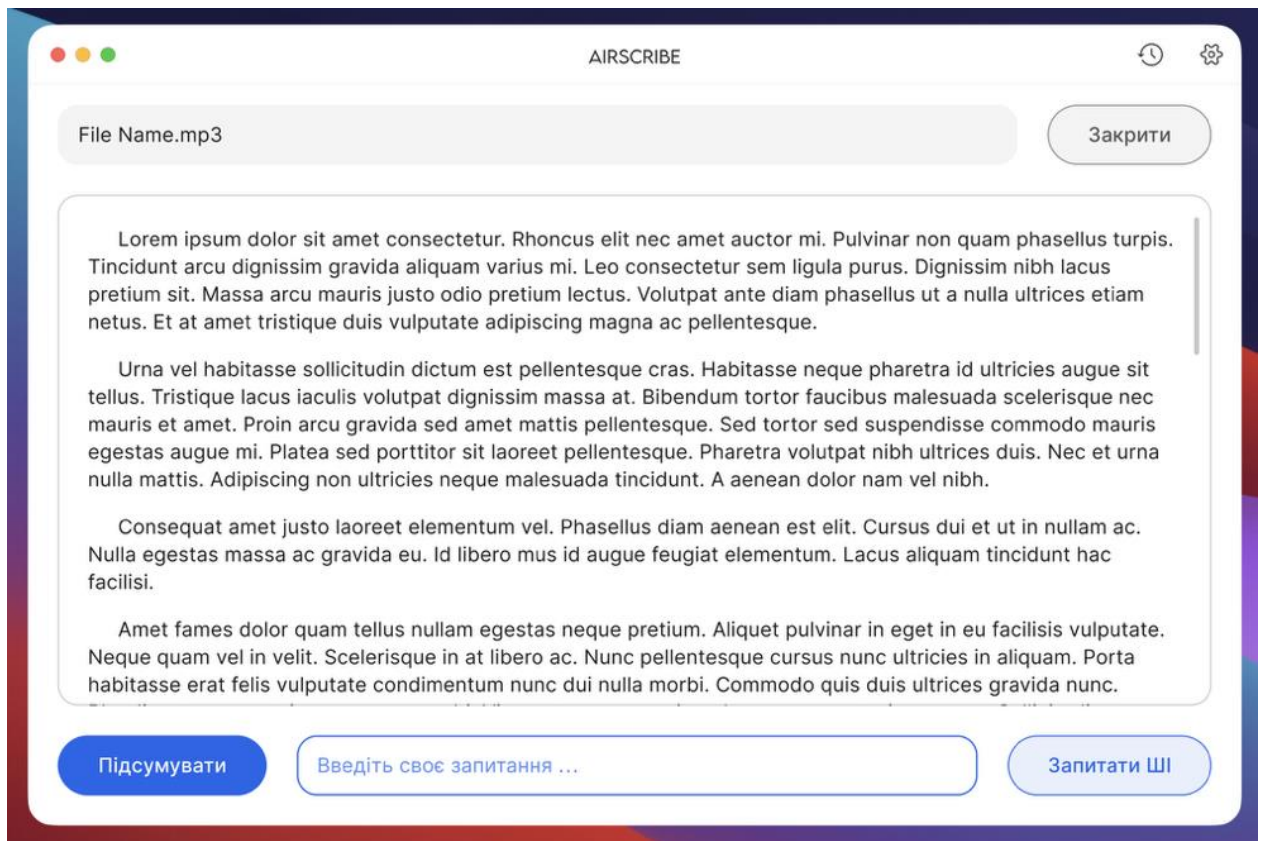


Рисунок 3.22 – Змінена структура тексту з поділом на абзаци

Одним з побажань покращення було додати кнопку копіювання результатів та підсумків. Ці опції було впроваджено в дизайні сторінок «Підсумки» та «Результат ШІ» та відображено на рисунку 3.23.

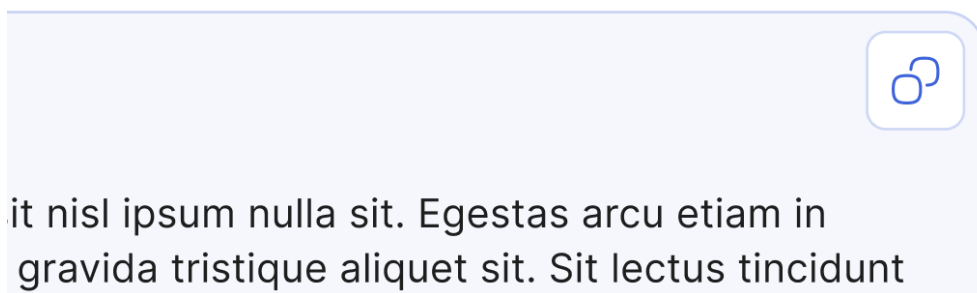


Рисунок 3.23 – Кнопка копіювання результату

Приклад екрану «Результат ШІ» з усіма врахованими та впровадженими коментарями та побажаннями було додано та продемонстровано на рисунку 3.24.

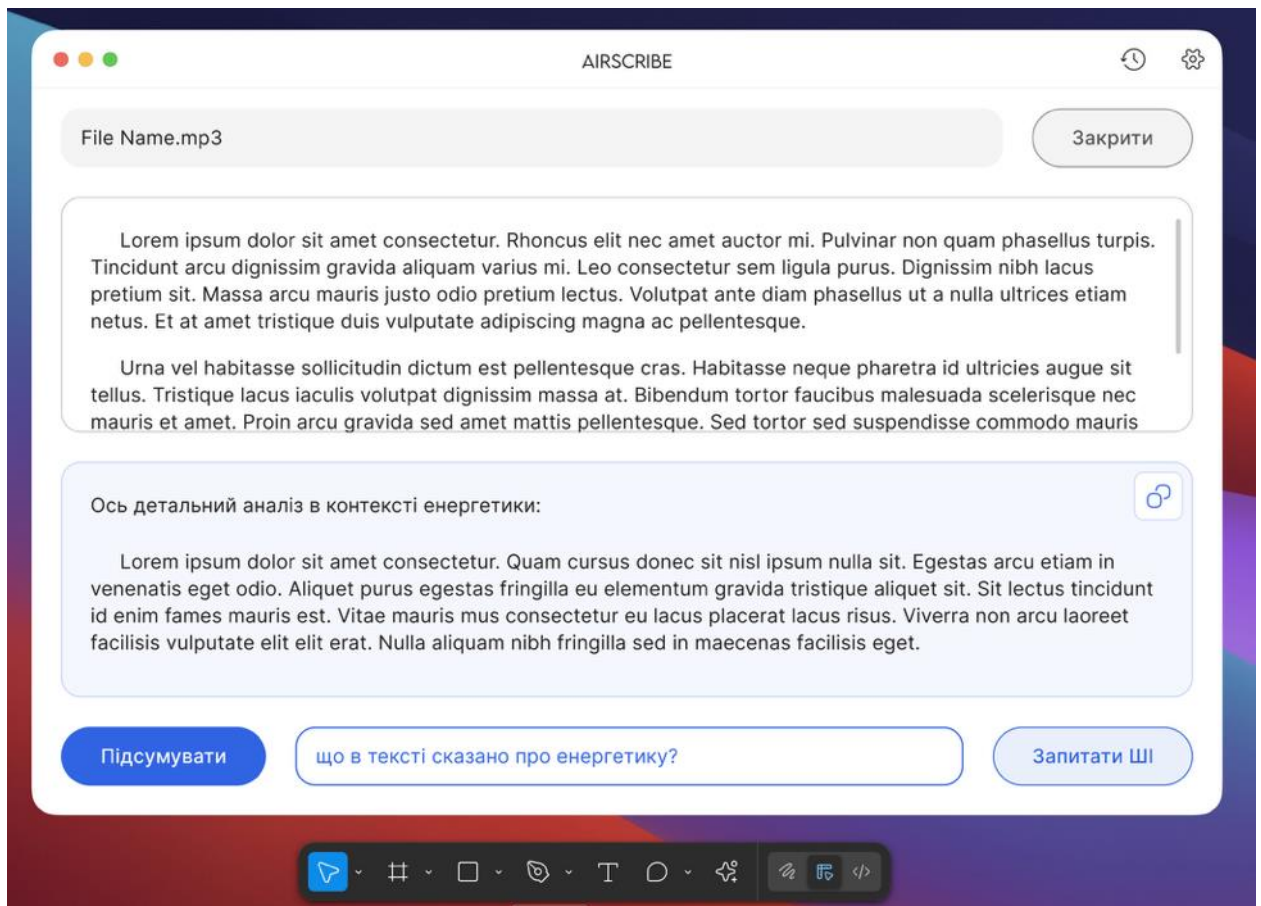


Рисунок 3.24 – Оновлений екран «Результат ШІ»

Усі внесені зміни стали результатом цілеспрямованої роботи з аналізу експертних відгуків і практичних побажань. Це дозволило не лише суттєво вдосконалити функціональні можливості інтерфейсу, а й досягти високого рівня зручності, логічності й інтуїтивної зрозумілості. Завдяки цьому користувач отримає передбачуваний і приємний досвід на всіх етапах взаємодії з застосунком – від завантаження файлу до роботи з результатами транскрипції.

4 ЕКОНОМІЧНА ЧАСТИНА

В рамках кваліфікаційної роботи було розроблено UI/UX дизайн інтерфейсу для локального застосунку транскрибування. Розробка якісного інтерфейсу є важливим етапом у створенні конкурентоспроможного програмного продукту, оскільки безпосередньо впливає на зручність, швидкість та ефективність роботи користувачів. Економічна доцільність даного проєкту полягає в мінімізації витрат користувачів на сторонні сервіси транскрипції, що потребують підключення до Інтернету та можуть не гарантувати максимальної безпеки обробки даних.

Перед початком робіт було розраховано витрати на розробку, включаючи витрати на робочий час, використання програмного забезпечення Figma, що має безкоштовний тарифний план для базових задач, а також витрати на дослідження та аналіз аудиторії. Основні переваги розробленого інтерфейсу полягають у його мінімалістичності, що дозволяє скоротити час навчання користувача, знизити кількість помилок при використанні та забезпечити високий рівень довіри до продукту завдяки акценту на локальній обробці даних.

Конкурентне середовище було проаналізовано з огляду на існуючі транскрипційні застосунки, що здебільшого працюють на основі хмарних сервісів. Основні переваги розробленого інтерфейсу порівняно з конкурентами полягають у:

- локальній обробці даних, що виключає ризик витоку конфіденційної інформації;
- простому та інтуїтивно зрозумілому інтерфейсі, орієнтованому на користувача без спеціальних технічних знань;
- мінімізації витрат користувача за рахунок відсутності необхідності в підключенні до мережі та підписок на платні сервіси.

Джерела економії для користувача – це економія на підписці на сторонні сервіси, а також зниження ризиків, пов'язаних із витоком даних. Для дизайнера витрати склалися з витрат часу на дослідження, створення прототипів, проведення UX-тестування, а також інших витрат, наприклад: обладнання. Джерелом фінансування виступали власні кошти дизайнера.

Розглянемо порядок розробки UI/UX дизайну застосунку для транскрибування. Цей процес включає наступні етапи:

- початковий етап, на якому здійснюється аналіз вимог, визначаються функціональні завдання застосунку, потреби цільової аудиторії та технічні обмеження продукту;
- етап проектування інтерфейсу, під час якого створюються wireframes, розробляється загальна логіка користувацького потоку та визначається основна ієрархія екранів;
- етап створення UI/UX-прототипів високої деталізації (High-Fidelity), що дозволяє перевірити зручність взаємодії та адаптивність інтерфейсу до різних сценаріїв використання;
- етап розробки інтерактивного прототипу, що дає змогу оцінити UX-логіку, виявити потенційні проблеми та отримати зворотній зв'язок користувачів та експертів ще до початку технічної реалізації продукту;
- етап, на якому вносяться зміни та доопрацювання на основі зібраного фідбеку, що забезпечує підвищення якості користувацького досвіду та готовність інтерфейсу до подальшої інтеграції в готовий застосунок.

Економічне обґрунтування проєкту включає оцінку витрат, пов'язаних із розробкою дизайну та збором зворотнього зв'язку. Основною статтею витрат виступає оплата праці одного UI/UX дизайнера, сплата єдиного соціального внеску, витрати на електроенергію та інші витрати.

У процесі розрахунку основної заробітної плати було враховано тривалість роботи дизайнера – 4 дні по 8 годин, зі ставкою – 240,00 грн/год. Розрахунок основної заробітної плати наведено в таблиці 4.1.

Таблиця 4.1 – Розрахунок витрат на заробітну плату

Етап	Вид робіт	Виконавець		Годинна ставка, грн	Тривалість виконання, дні	Заробітна плата, грн
		к-ть, ос.	посада			
1. Початковий	Визначення функціональних вимог	1	UI/UX Дизайнер	240,00	0,5	960,00
2. Проектування інтерфейсу	Створення прототипів wireframes	1	UI/UX Дизайнер	240,00	1	1920,00
3. Створення UI/UX прототипів	Створення макетів дизайну High-Fidelity	1	UI/UX Дизайнер	240,00	1	1920,00
4. Створення інтерактивного прототипу	Поєднання прототипів в інтерактивний досвід та перевірка зручності інтерфейсу	1	UI/UX Дизайнер	240,00	1	1920,00
5. Коригування дизайну	Внесення змін в дизайн, спираючись на отриманий зворотній зв'язок	1	UI/UX Дизайнер	240,00	0,5	960,00
Усього						7680,00

Ставка єдиного соціального внеску становить 22% від величини основної і додаткової заробітної плати:

$$7680,00 \times 0,22 = 1689,60 \text{ грн.}$$

До інших витрат відносяться витрати на обслуговування комп'ютерної техніки, що використовується виконавцем проекту, і плату за електроенергію. Витрати на електроенергію розраховуються виходячи зі споживаної потужності пристрою і тарифу на електроенергію. У даному випадку

передбачається використання одного комп'ютера з потужністю 0,8 кВт/год. Вартість 1 кВт/год електроенергії прийнято у розмірі 4,32 грн. Час використання електроенергії в процесі розробки:

$$8 \times 4 = 32 \text{ год.}$$

Тобто плата за спожиту електроенергію становитиме:

$$0,8 \times 4,32 \times 32 \times 1 = 110,60 \text{ грн.}$$

Витрати на обслуговування техніки визначаються виходячи з її вартості та часу експлуатації, після закінчення якого, вона підлягає заміні. Отже, враховуючи, що вартість комп'ютера необхідного для такого проекту 70000,00 грн, а протягом року техніка використовується 254 робочих дні, отримаємо наступну суму витрат на обслуговування за час виконання проекту:

$$(70000,00 / (3 \times 8 \times 254)) \times 32 = 367,46 \text{ грн.}$$

Собівартість розробки становить:

$$(7680,00 + 1689,60 + 110,60 + 367,46) / 1 = 9847,66 \text{ грн.}$$

Розрахуємо суму прибутку від реалізації розробки (виходячи з рівня рентабельності 30 %):

$$9847,66 \times 0,3 = 2954,30 \text{ грн.}$$

Розрахуємо ціну розробки дизайну без податку на додану вартість (ПДВ):

$$9847,66 + 2954,30 = 12801,96 \text{ грн.}$$

Розрахуємо суму ПДВ, що дорівнює 20 % від ціни без ПДВ:

$$12801,96 \times 0,2 = 2560,39 \text{ грн.}$$

З урахуванням проведених розрахунків ціна розробки дизайну інтерфейсу для застосунка з ПДВ складає:

$$12801,96 + 2560,39 = 15362,35 \text{ грн.}$$

Результати розрахунків наведено у таблиці 4.2.

Таблиця 4.2 – Розрахунок витрат на розробку дизайну застосунку

№ з/п	Стаття витрат	Сума, грн
1	Основна заробітна плата	7680,00
2	Єдиний соціальний внесок	1689,60
3	Витрати на електроенергію.	110,60
4	Витрати на обслуговування техніки	367,46
5	Собівартість розробки сайту	9847,66
6	Прибуток	2954,30
7	Ціна без ПДВ	12801,96
8	Податок на додану вартість (ПДВ)	2560,39
9	Ціна з урахуванням ПДВ	15362,35

Отже, повна вартість розробки UI/UX дизайну для локального застосунку транскрибування складає 15362,35 грн. Виконання всіх стадій дизайнером триватиме 4 дні. Очікуваний прибуток складає 2954,30 грн, , що свідчить про доцільність розробки.

ВИСНОВКИ

У ході виконання кваліфікаційної роботи було створено прототип інтерфейсу локального офлайн-застосунку для транскрибування мовлення зі вбудованим штучним інтелектом. Проєкт реалізовано відповідно до поставлених цілей і задач, із урахуванням актуальних вимог до безпеки, автономності та зручності використання.

В рамках роботи було проаналізовано проблематику існуючих транскрипційних рішень. Визначено цільову аудиторію – користувачі, що працюють із чутливими аудіоматеріалами (журналісти, підприємці, викладачі). Проведено аналітичний огляд сучасних застосунків транскрибування, занотовано їхні позитивні критерії, які бажано імплементувати в створюваний застосунок. Досліджено сучасні методики UX дизайну – Design Thinking, яка складається з: емпатії, визначення проблеми, ідеації, прототипування та тестування, та Human-Centered Design, яка ставить досвід та емоції користувача перед усім. Застосування цих методик на практиці допомогло ефективно створити прототип інтерфейсу, для побудови якого було проаналізовано інструменти для створення інтерактивних прототипів.

Було продумано сценарії взаємодії користувача, для досягнення логічності та послідовності інтерфейсу. Побудовано структурну схему для відображення функціональних можливостей та створено UML діаграму для подальшого узгодження з дизайном прототипу. На основі побудованих діаграм взаємодії розроблено прототипи-схеми wireframes за допомогою інструменту для прототипування та створення дизайну Figma.

Спираючись на схеми wireframes, було розроблено дизайн інтерфейсу з логічним доступом до основних функцій: завантаження файлу, транскрибування, підсумовування, звернення до ІІІ, налаштувань та історії. Приділено додаткову увагу на додавання елементів дизайну для функцій безпеки, як шифрування та повне видалення інформації.

Після чого екрани дизайну було поєднано зв'язками взаємодії у клікабельний прототип для тестування. Було проведено тестування із залученням експертів з різних галузей. Отримано та проаналізовано їхній зворотний зв'язок щодо рішень в дизайні інтерфейсу застосунку.

Спираючись на відгуки та пропозиції щодо покращення було внесено зміни в дизайн прототипу, які зробили інтерфейс більш інтуїтивним і передбачуваним для кінцевого користувача.

Таким чином, у результаті виконання кваліфікаційної роботи було реалізовано повноцінний інтерфейс захищеного офлайн-застосунку зі створення транскрипцій, який відповідає сучасним вимогам зручності, функціональності та конфіденційності даних.

ПЕРЕЛІК ДЖЕРЕЛ ПОСИЛАННЯ

1. Методичні вказівки з виконання кваліфікаційної роботи для студентів денної та заочної форми навчання першого (бакалаврського) рівня вищої освіти спеціальності 186 «Видавництво та поліграфія» за освітньою програмою «Видавничо-поліграфічна справа» / В.П. Ткаченко, А.В. Бізюк, О.В. Вовк, І.М. Єгорова, В.Ф. Челомбійко. Харків: ХНУРЕ, 2020. 68 с.
2. Полозова Т.В. Методичні вказівки до виконання економічної частини кваліфікаційної роботи для здобувачів першого (бакалаврського) рівня вищої освіти спеціальності 186 Видавництво та поліграфія усіх форм навчання. Харків: ХНУРЕ, 2022. 47 с.
3. ДСТУ 8302:2015. Інформація та документація. Бібліографічне посилання. Загальні положення та правила складання. Київ, 2016. 16 с.
4. ДСТУ 3008:2015. Інформація та документація. Звіти у сфері науки і техніки. Структура та правила оформлювання. Київ, 2016. 31 с.
5. Жидецький В.Ц., Лях І.М. Виробничі фактори, що чинять несприятливу дію на оператора комп'ютерного набору // Поліграфічні, мультимедійні та web-технології. 2025. Т. 1. С. 23 24.
6. Unveiling the power of speech-to-text algorithms // Verbit. URL: <https://verbit.ai/ai-technology/speech-to-text-algorithms/> (дата звернення: 30.05.2025).
7. Левикін І.В., Шимко Д.І. Сучасні інструменти автоматизованої конверсії відеоконтенту в статичні зображення та текстову інформацію // Поліграфічні, мультимедійні та web-технології. 2025. Т. 1. С. 110-111.
8. Григор'єв О.В., Вовк О.В. Засоби обробки інформації для мультимедійних продуктів // Поліграфічні, мультимедійні та web-технології. 2025. Т. 1. С. 195 196.
9. XChaCha20 encryption vs. AES-256: what's the difference? // NordPass. URL: <https://nordpass.com/blog/xchacha20-encryption-vs-aes-256/> (дата звернення: 30.05.2025).
10. Rams D. Less and More / ред.: К. Klemp, К. Ueki-Polet. 2015. 808 с.

11. Настанови з доступності вебвмісту (WCAG) 2.1. W3C. URL: <https://www.w3.org/Translations/WCAG21-ua/> (дата звернення: 30.05.2025).
12. What is Design Thinking? – updated 2025. The Interaction Design Foundation. URL: <https://www.interaction-design.org/literature/topics/design-thinking> (дата звернення: 30.05.2025).
13. Human-centered design: як, коли та навіщо обирати цей дизайн-процес. Computools. URL: <https://careers.computools.ua/human-centered-design-how-when-and-why/> (дата звернення: 30.05.2025).
14. Config 2025: pushing design further // Figma. URL: <https://www.figma.com/blog/config-2025-recap/> (дата звернення: 30.05.2025).
15. Григор'єв О.В., Вовк О.В., Виволокін А.О. Засоби отримання інформації для мультимедійних продуктів // Поліграфічні, мультимедійні та web-технології. 2025. Т. 1. С. 186-187.
16. Як визначити цільову аудиторію // IdeaDigital Agency. URL: https://ideadigital.agency/blog/yak-viznachiti-tsilovu-auditoriyu/#list_title5 (дата звернення: 30.05.2025).
17. Діаграми UML для моделювання процесів і архітектури проекту. Evergreen – web розробка і діджиталізація бізнесу за допомогою AI продуктів. URL: <https://evergreens.com.ua/ua/articles/uml-diagrams.html> (дата звернення: 30.05.2025).
18. M B. Principles of visual hierarchy in UI Design // Medium. URL: <https://uxplanet.org/principles-of-visual-hierarchy-in-ui-design-fbcd31f88088> (дата звернення: 30.05.2025).
19. Desktop screen resolution stats worldwide // Statcounter. URL: <https://gs.statcounter.com/screen-resolution-stats/desktop/worldwide> (дата звернення: 30.05.2025).
20. Ukrainian W. Уроки від W3Schools українською онлайн // W3Schools українською. Безплатні уроки онлайн для студентів. URL: https://w3schoolsua.github.io/html/html5_draganddrop.html#gsc.tab=0 (дата звернення: 30.05.2025).
21. Guide to components in Figma. Figma Learn // Help Center. URL: <https://help.figma.com/hc/en-us/articles/360038662654-Guide-to-components-in-Figma> (дата звернення: 30.05.2025).

22. What Are High-Fidelity Prototypes?. The Interaction Design Foundation. URL: <https://www.interaction-design.org/literature/topics/high-fidelity-prototypes> (дата звернення: 30.05.2025).

23. Understanding Color Theory of Blue & Its Use Cases | AND Academy. AND Academy. URL: <https://www.andacademy.com/resources/blog/graphic-design/color-theory-blue/> (дата звернення: 30.05.2025).

24. Figma 101: Create and Test a Prototype | Designlab. Online UI and UX Design Courses and Bootcamps // Designlab. URL: <https://designlab.com/figma-101-course/create-and-test-a-prototype> (дата звернення: 30.05.2025).