

Харківський національний університет радіоелектроніки

Факультет _____ комп'ютерної інженерії та управління _____

Кафедра _____ електронних обчислювальних машин _____

Рівень вищої освіти _____ перший (бакалаврський) _____

Спеціальність _____ 123 «Комп'ютерна інженерія» _____
(код і повна назва)

Тип програми _____ освітньо-професійна _____
(освітньо-професійна або освітньо-наукова)

Освітня програма _____ Комп'ютерна інженерія _____
(повна назва)

ЗАТВЕРДЖУЮ:

Зав. кафедри _____
(підпис)

“ _____ ” _____ 20__ р.

ЗАВДАННЯ

НА КВАЛІФІКАЦІЙНУ РОБОТУ

здобувачеві _____ Потапову Ростиславу Олександровичу _____
(прізвище, ім'я, по батькові)

1. Тема роботи Програмний модуль «Розпізнавання та автопунктуація тексту»

затверджена наказом по університету від “ 26 ” травня 2025 р. № 424 Ст

2. Термін подання здобувачем роботи до екзаменаційної комісії 17 червня 2025 р.

3. Вхідні дані до роботи 1) нормативно-правові та законодавчі акти України;

2) ДСТУ щодо обробки інформації;

3) періодичні видання;

4) науково-методичні розробки вітчизняних та зарубіжних авторів, матеріали;

4. Перелік питань, що потрібно опрацювати у роботі _____

1) аналіз проблеми та огляд існуючих рішень;

2) вибір технології розробки та інструментальних засобів;

3) розробка алгоритмічного забезпечення;

4) розробка програмних модулів;

5) відлагодження програмних модулів;

6) висновки.

5. Перелік графічного матеріалу із зазначенням креслеників, схем, плакатів, комп'ютерних ілюстрацій _____

Слайд-презентація – 20 слайдів _____

6. Консультанти розділів роботи (заповнюється 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

КАЛЕНДАРНИЙ ПЛАН

№	Назва етапів роботи	Строк / терміни виконання етапів роботи	Примітка
1	Аналіз проблеми та огляд існуючих рішень	26.05.25-30.05.25	
2	Вибір технології розробки та інструментальних засобів	31.05.25-02.06.25	
3	Розробка алгоритмічного забезпечення	03.06.25-05.06.25	
4	Розробка та відлагодження програмного забезпечення	06.06.25-09.06.25	
5	Оформлення матеріалів кваліфікаційної роботи	10.06.25-11.06.25	
6	Подання кваліфікаційної роботи керівникові та її попередній захист	12.06.25-13.06.25	
7	Подання кваліфікаційної роботи на рецензування	14.06.25-16.06.25	

Дата видачі завдання “ 26 ” травня 2025 р.

Здобувач _____

(підпис)

Керівник роботи _____

(підпис)

доц. Володимир ФЕДОРЧЕНКО _____

(посада, власне ім'я, прізвище)

РЕФЕРАТ

Пояснювальна записка кваліфікаційної роботи: 76 с., 25 рис., 28 табл., 41 джерело.

АВТОПУНКТУАЦІЯ, ЗАПИС МОВИ, РОЗПІЗНАВАННЯ, BERT, FIREBASE, SPEECHMATICS, SWIFT.

Метою кваліфікаційної роботи є створення мобільного додатку з функціями розпізнавання та автопунктуації тексту.

Об'єктом кваліфікаційної роботи є функціональні елементи, архітектура, програмне забезпечення мобільного додатку та модулі розпізнавання та автопунктуації тексту.

Метод проектування – використання програмної системи Xcode, мови програмування Swift, технологій розпізнавання Speechmatics. База даних – Firebase.

Створений мобільний додаток може бути застосовано для запису ділових розмов, конференцій, допитів, запису розмов під час медичних операцій.

Результати роботи можуть бути впроваджені у будь-якій сфері, у якій є необхідність роботи з великими обсягами даних.

ABSTRACT

The explanatory note of the qualification work's thesis report: 76 pages, 25 figures, 28 tables, 41 sources.

AUTOPUNCTUATION, BERT, FIREBASE, RECOGNITION, RECORD SPEECH, SPEECHMATICS, SWIFT.

The objects of designing are functional elements, architecture, software of mobile application and modules of recognition and auto-punctuation of text.

The purpose of designing is to create a mobile application with features of recognition and auto-punctuation of text.

The method of designing – using the Xcode software system, Swift programming language, Speechmatics recognition technology. Database – Firebase.

The created mobile application can be used for recording business conversations, conferences, interrogations, recording of conversations during medical operations.

The results of the work can be implemented in any field with necessity to work with large amounts of data.

ЗМІСТ

СКОРОЧЕННЯ ТА УМОВНІ ПОЗНАКИ	8
ВСТУП	9
1 АНАЛІЗ ПРЕДМЕТНОЇ ОБЛАСТІ ОБРОБКИ МОВИ.....	11
1.1 Основні принципи розпізнавання та автопунктуації тексту	11
1.2 Опис предметної області	12
1.3 Огляд і аналіз наявних аналогів, що реалізують функції предметної області	16
2 СПЕЦИФІКАЦІЯ ВИМОГ ДО СИСТЕМИ	20
2.1 Глосарій.....	20
2.2 Розроблення варіантів використання.....	21
2.2.1 Діаграма варіантів використання	21
2.2.2 Специфікація варіантів використання	22
2.3. Специфікація функціональних та нефункціональних вимог	27
2.4. Проектування інтерфейсу користувача	30
3. ПРОЄКТНІ ТА ТЕХНІЧНІ РІШЕННЯ	33
3.1. Логічна постановка	33
3.2. Проектування структури бази даних	35
3.2.1. Концептуальне інфологічне проектування	35
3.2.2 Проектування логічної моделі даних.....	36
3.2.3. Проектування фізичної моделі даних	37
3.3. Проектування програмного забезпечення	37
3.4. Тестування програмної системи.....	44
3.5. Розгортання програмного продукту.....	51
ВИСНОВКИ.....	53
ПЕРЕЛІК ДЖЕРЕЛ ПОСИЛАННЯ	54
ДОДАТОК А Графічний матеріал кваліфікаційної роботи.....	58
ДОДАТОК Б Лістинг програми.....	69

Б.1 Код інтерфейсу та логіки для створення аудіо.	69
Б.2 Код інтерфейсу та логіки для розпізнавання та трансформування аудіо.	71
Б.3 Сервіс роботи з базою даних.	74

СКОРОЧЕННЯ ТА УМОВНІ ПОЗНАКИ

СУБД – система управління базами даних

BERT (англ. Bidirectional Encoder Representations from Transformers) -
двонаправлені представлення кодерів від трансформаторів

DFD (англ. data flow diagrams) — діаграми потоків даних

IDEF – методологія функціонального моделювання і графічного
описання процесів

LSTM (англ. long short-term memory) – довга короткочасна пам'ять

NLP (англ. Natural language processing) – обробка природної мови

RNN (англ. recurrent neural network) – рекурентна нейронна мережа

UI (англ. user interface) – інтерфейс користувача

UML (англ. Unified Modeling Language) – уніфікована мова
моделювання

ВСТУП

Метою кваліфікаційної роботи є проектування та створення мобільного застосунку для обробки людської мови.

У сучасному світі людство постійно взаємодіє з великою кількістю інформації, деяка інформація може знадобитися декілька разів, тому має бути збережена. Збереження інформації може бути проблемою, якщо йдеться про звуковий формат. Наприклад, записи ділових розмов, конференцій, допитів, записи розмов під час медичних операцій можуть займати багато дискового простору та робота з ними є незручною у разі потреби у пошуку якої-небудь деталі, але робота з текстовим форматом є набагато зручнішою, тому що не потребує багато простору для зберігання та усі текстові редактори мають функцію пошуку. Також текстовий формат представлення даних можна використати для оформлення звітів, статей та протоколів, або при необхідності перекласти на іншу мову.

Отже, рішенням цієї проблеми є додаток, що має можливість запису мови, перетворення її у текст з подальшою автопунктуацією. Рішення реалізоване для iOS платформи, а пунктуацію буде виконано для англійської мови. Також передбачено систему автентифікації користувача та збереження його аудіозаписів у базі даних.

На сьогоднішній день вже існують аналоги, але зазвичай вони реалізовані у вигляді web-застосунку або десктопної версії, що не надає гнучкості. Також задача автопунктуації зазвичай вирішується шляхом вимови пунктуаційних команд, тому ці рішення не можуть бути застосовані до природної мови.

Для розробки мобільного додатку використано середу розробки Xcode, а для запису аудіо нативна iOS бібліотека AVFoundation. Для розпізнавання аудіо застосовано Speechmatics API, що також дозволяє поділяти аудіо на смислові сегменти, а для автопунктуації – API, що використовує модель

BERT. Базою даних є Firebase, що також має систему автентифікації.

Для проектування додатку використано інструменти Rational Rose та Ramus для створення UML та IDEF діаграм.

Результати кваліфікаційної роботи можуть бути використані у будь-якій галузі, що працює з великою кількістю інформації та документообігом.

1 АНАЛІЗ ПРЕДМЕТНОЇ ОБЛАСТІ ОБРОБКИ МОВИ

1.1 Основні принципи розпізнавання та автопунктуації тексту

Розпізнавання тексту [10] включає кілька етапів: попередню обробку, сегментацію, класифікацію символів і постобробку (рисунок 1.1).

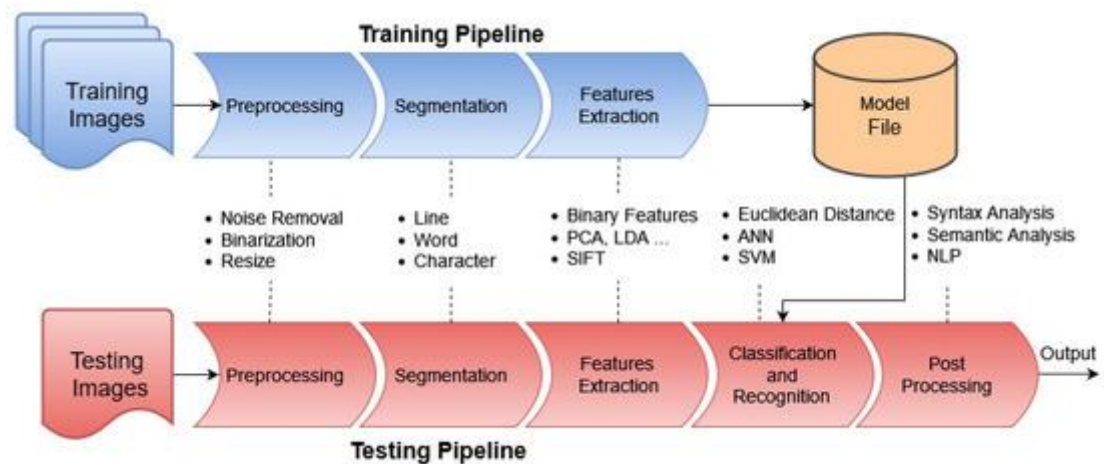


Рисунок 1.1 – Схема процесу розпізнавання рукописного тексту

Традиційні методи засновані на алгоритмах обробки зображень, таких як бінаризація, видалення шуму, виділення контурів та сегментація символів, методи порівняння шаблонів дозволяють розпізнавати символи, але погано працюють з рукописним текстом та різними шрифтами.

Методи машинного навчання включають класичні алгоритми (SVM, Random Forest) які використовуються для класифікації окремих символів, глибокі нейромережі (CNN, RNN), що дозволяють аналізувати зображення тексту як цілісний об'єкт.

В теперішній час активно розвиваються сучасні нейромережові підходи, а саме: CRNN (поєднання згорткових та рекурентних мереж) – успішно застосовується в сучасних OCR-системах; Transformer-моделі, такі як TrOCR, що дозволяють розпізнавати текст у складних умовах (розмитість,

спотворення) [3].

Автоматичне розміщення розділових знаків вимагає розуміння контексту, структури речення та мовних закономірностей [5].

Для визначення правил та лінгвістичних моделей застосовується аналіз морфології та синтаксису, що дозволяє виявляти місця, де можливі коми, крапки та інші знаки. Використовуються парсери та POS-тегери – для визначення структури речення.

Для визначення структури та мовних закономірностей застосовують статистичні методи та моделі: N-грамні моделі – аналізують послідовності слів та обчислюють ймовірність постановки знака; байєсовські методи використовують ймовірнісні оцінки на основі великих текстових корпусів.

Нейромережові підходи, а саме RNN, LSTM, GRU застосовуються для передбачення пунктуації на основі аналізу послідовностей. Transformer-моделі (BERT, T5, GPT) забезпечують високу точність, враховуючи контекст цілого документа. Sequence-to-sequence моделі вирішують завдання у форматі машинного перекладу.

Сучасні системи часто поєднують традиційні методи з нейромережами. Наприклад, спочатку проводиться попередня розмітка тексту за допомогою правил, потім застосовується нейромережева модель для уточнення результатів.

1.2 Опис предметної області

Є багато інформації, що потребує подальшого зберігання та, наприклад, обробки та оновлення, але збереження інформації може бути проблемою, якщо йдеться про звуковий формат. Наприклад, записи ділових розмов, конференцій, допитів, записи розмов під час медичних операцій можуть займати багато дискового простору та робота з ними є незручною у разі потреби у пошуку якої-небудь деталі, але робота з текстовим форматом є набагато зручнішою, тому що не потребує багато простору для зберігання та

усі текстові редактори мають функцію пошуку. Також текстовий формат представлення даних можна використати для оформлення звітів, статей та протоколів, або при необхідності перекласти на іншу мову.

Тому є необхідність створення застосунку, що буде мати можливість запису мови з подальшим перетворенням у текст та виконанням автопунктуації. Рішення має бути реалізовано у вигляді мобільного додатку, тому що це надає можливість більш гнучкого використання, а оскільки рішення проектується для iOS платформи, то у разі необхідності у майбутньому додаток може бути з легкістю портований на платформу MacOS.

IDEF0 – методологія функціонального моделювання [33]. За допомогою графічної мови IDEF0, задача постає перед розробниками та аналітиками у вигляді набору взаємозалежних функцій. Як правило, моделювання засобами IDEF0 є першим етапом вивчення будь-якої системи.

Відображення структури системи, у якій зв'язками є інформаційні потоки, також можна здійснити за допомогою діаграм потоків даних DFD [9]. Ці діаграми використовують для аналізу та моделювання інформаційних систем з метою мінімізації потоків даних та зменшення їх об'єму, виявлення як дублювання інформації, так і дублювання шляхів її передавання. DFD відображають джерела та споживачів інформації, вид та напрямок передачі інформації, елементи накопичення та процеси перетворення, при цьому використовуються різні засоби відображення елементів.

DFD є основним засобом моделювання функціональних вимог проєктованої системи. З їх допомогою ці вимоги розбиваються на функціональні компоненти і представляються у вигляді мережі, пов'язаної потоками даних.

Головна мета таких засобів – продемонструвати, як кожен процес перетворить свої вхідні дані у вихідні, а також виявити відносини між цими процесами (таблиця 1.1, рисунок 1.2 – 1.4).

Таблиця 1.1 – Характеристика бізнес процесу «Перетворення повідомлення»

Назва характеристики	Значення характеристики
Ім'я бізнес-процесу	Перетворення повідомлення
Основні учасники	Учасниками процесу є усі працівники об'єкту управління
Вхідна подія	Засідання, виступ, семінар
Вхідні документи	Аудіозапис засідання, виступу, семінару
Вихідна подія	Використання документу у подальшому документооберті
Вихідні документи	Печатне представлення аудіозапису
Клієнт бізнес-процесу	Документооберт

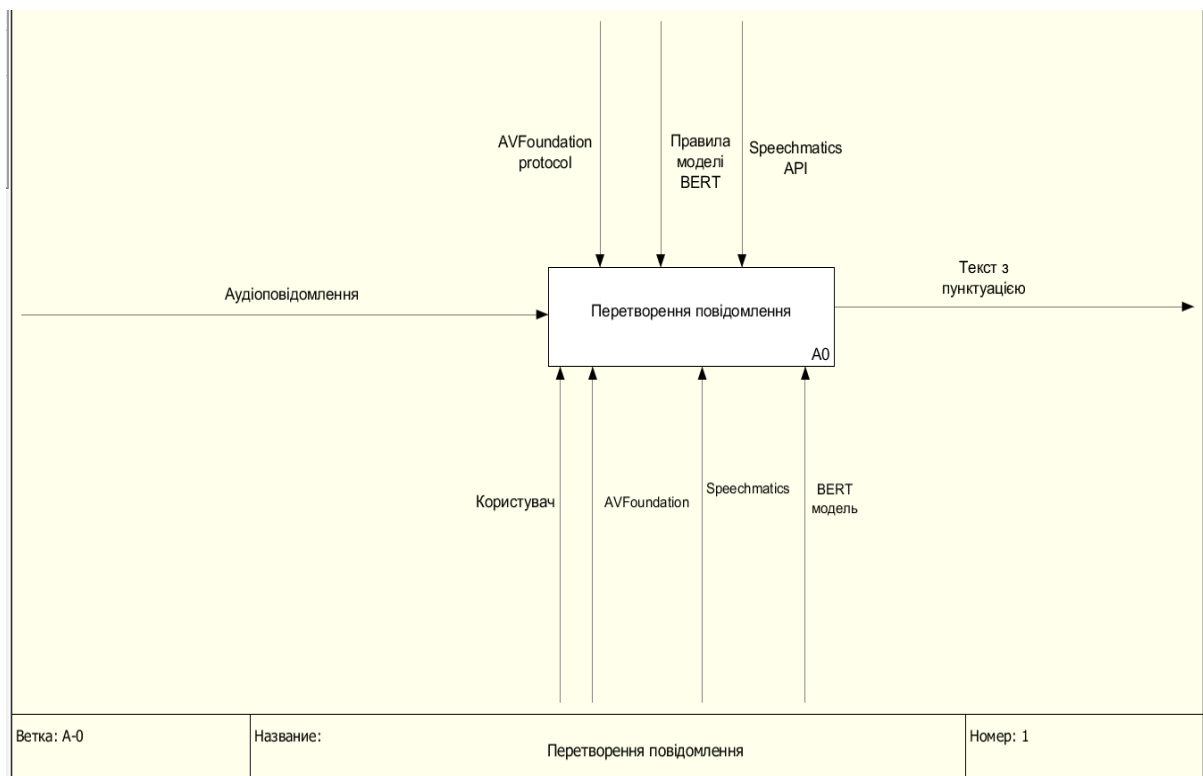


Рисунок 1.2 – Діаграма IDEF0, рівень 0

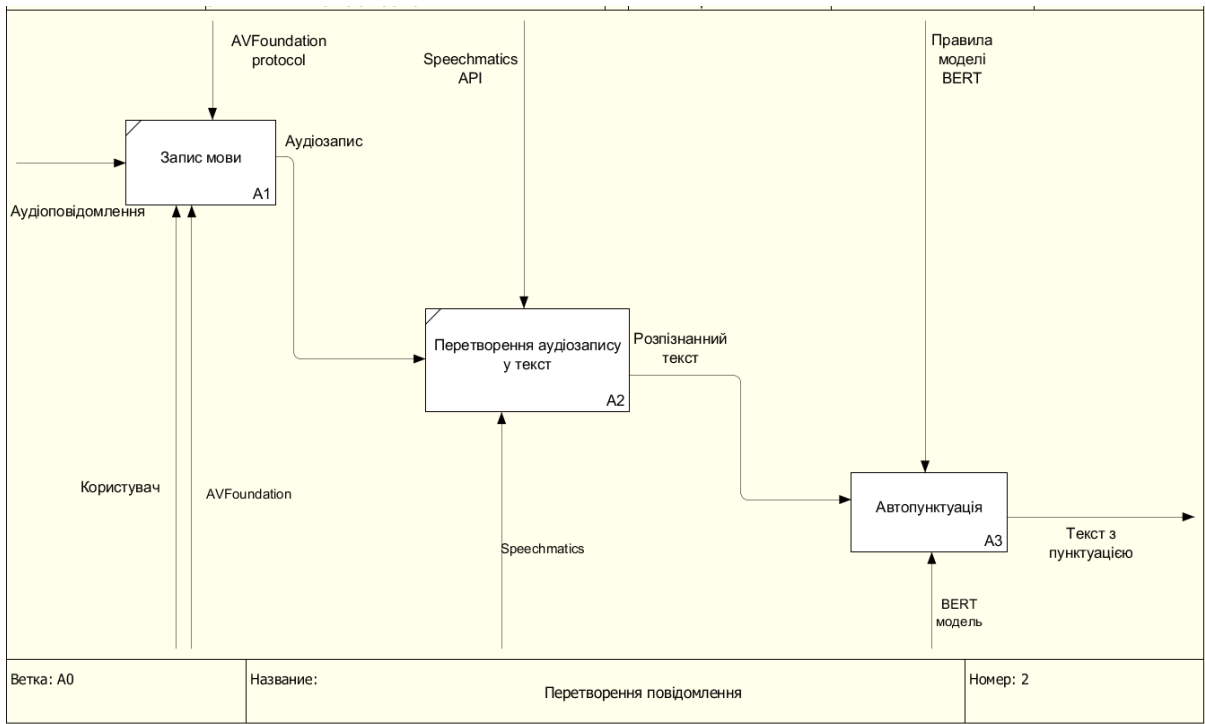


Рисунок 1.3 – Діаграма IDEF0, рівень 1

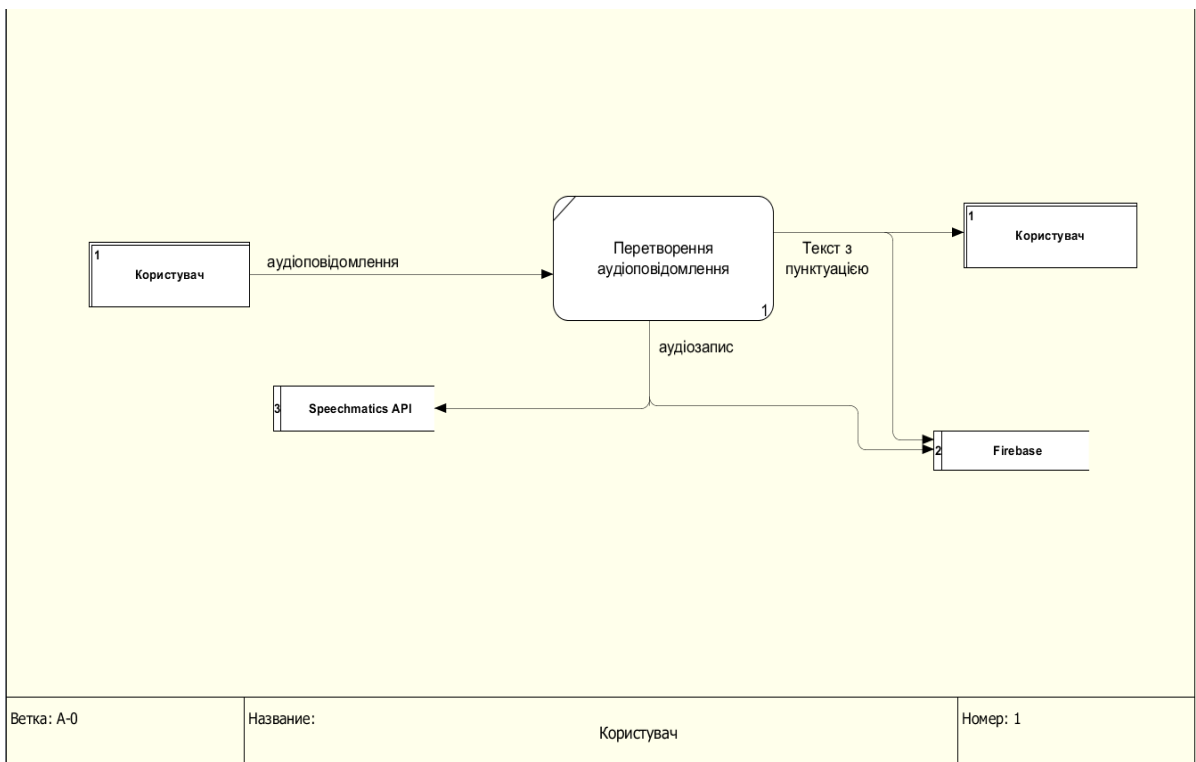


Рисунок 1.4 – Діаграма DFD

1.3 Огляд і аналіз наявних аналогів, що реалізують функції предметної області

На сьогоднішній день є багато сервісів, що дозволяють голосовий набір тексту, але значна більшість з них реалізують пунктуацію за рахунок вимови знаків, що потребує багато зайвого часу і тому знижує ефективність використання. Тобто вони не можуть бути використані для вільного запису розмови, допиту, тощо. Також таке рішення допомагає людям з обмеженими можливостями, але відсутність автопунктуації є значним недоліком.

Ще одним зауваженням є те, що багато таких рішень реалізовано у вигляді web-застосунків, використання яких не є гнучким та завжди потребує інтернету (таблиця 1.2).

На рисунку (рисунок 1.5 – 1.8) наведені скріншоти обраних аналогів








Рисунок 1.5 – VoiceNote

FIND WORK | MY WORK | **QUALIFICATIONS** | RATING | SUPPORT CENTER | SEND FEEDBACK

GET QUALIFIED & GET TO WORK

The number of jobs available to you depends upon your Qualifications and your Rating. More Qualifications equals more work, pretty simple, huh? Get qualified now!

QUALIFICATIONS

- 
Basic
 The Basic Qualification is required before you can transcribe any job. Take this test now and get started!
QUALIFICATION GRANTED: 09/02/2016 6:09 PM
- 
Confidentiality
 The Confidentiality Qualification is required for certain jobs that have strict rules regarding the confidentiality of the content.
QUALIFICATION GRANTED: 09/02/2016 6:11 PM
- 
Home Inspection Audio
 We're currently not hiring for this qualification. Please check back in 6 months.
 Transcribe a home inspection audio file.
QUALIFICATION NOT AVAILABLE
- 
Reviewer
 Invite Only - The Reviewer Qualification is available only to workers who have transcribed 1,000 minutes of audio while maintaining a rating of 96 or higher. This qualification tests your ability to listen to an audio or video file and review a transcription created by another transcriber.
70 CURRENT JOBS REQUIRE THIS QUALIFICATION
- 
Rush Job
 Invite Only - The Rush Jobs Qualification is available only to workers who have transcribed 500 minutes of audio while maintaining a rating of 96 or higher.
89 CURRENT JOBS REQUIRE THIS QUALIFICATION

DID YOU KNOW?

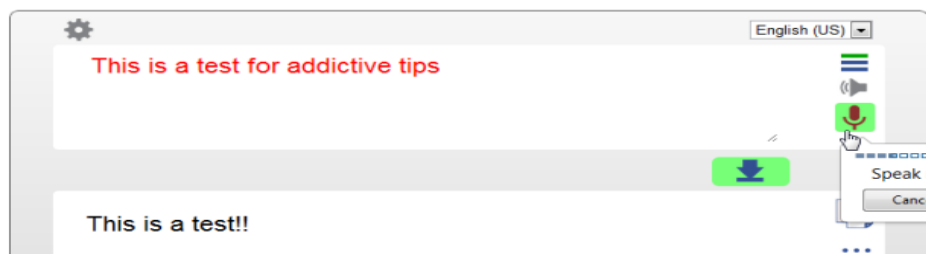
Qualifications & Ratings work together to determine which jobs are available to you.

[LEARN MORE](#)

Рисунок 1.6 – Speechpad



Free speech to text dictation software in a browser.



The screenshot shows a web browser window with the TalkTypewriter interface. At the top right, there is a language dropdown menu set to "English (US)". Below it, there are icons for volume and a microphone. A red microphone icon is highlighted, and a context menu is open over it, showing "Speak n" and "Cancel" options. The main text area contains the sentence "This is a test for addictive tips" in red. Below the text area, there is a green download icon and the text "This is a test!!".

Рисунок 1.7 – TalkTypер



Рисунок 1.8 – Voco

Таблиця 1.2 – Порівняльна характеристика аналогів

Фірма-розробник	Google Chrome	Simplify Now, LLC	TalkTyper	ЦРТ
Назва програмного продукту	Speechpad	VoiceNote	TalkTyper	Voco
Версії продукту	Web	Web, Mobile	Web	Desktop, Windows 7+
Функціональність	Набір та редагування тексту голосом, перетворення аудіо файлів у текст, синхронний переклад	Набір та редагування тексту голосом, перетворення аудіо файлів у текст, налаштування інтерфейсу	Набір та редагування тексту голосом	Набір та редагування тексту голосом, транскрибування аудіо, також можна розширити версію словниками (наприклад, з юридичними термінами або іншими)
Автопунктуація	Пунктуація виконується у разі вимови відповідних команд	Пунктуація додається у режимі редагування власноруч	Пунктуація виконується у разі вимови відповідних команд	Пунктуація виконується у разі вимови відповідних команд, але при розпізнаванні аудіо файлів пунктуація виконується автоматично

Продовження таблиці 1.2

Переваги	Є багато налаштувань та мов	Є багато додаткових налаштувань інтерфейсу	Голосові підказки, можливості перегляду варіантів розпізнавання	Великий функціонал, працює оффлайн
Недоліки	Працює тільки у Google Chrome та за наявності інтернет.	Розпізнавання тексту працює в онлайн режимі, але деякі налаштування можна виконувати оффлайн	Невеликий функціонал	Тільки платна версія

2 СПЕЦІФІКАЦІЯ ВИМОГ ДО СИСТЕМИ

2.1 Глосарій

Глосарій проекту – словник до тексту, що пояснює маловідомі або застарілі слова, що досліджується чи описується з тлумаченням, іноді перекладом на іншу мову, коментарями і прикладами [15]. Глосарій наведено у таблиці (таблиця 2.1).

Таблиця 2.1 – Глосарій

Термін	Опис терміну
1. Основні поняття та категорії предметної області та проекту	
Аккаунт	Обліковий запис користувача, що доступний тільки йому та надає доступ до створених ним матеріалів.
Матеріали	Об'єкти, що створені користувачем: аудіозаписи, їх текстове представлення.
Статус аудіозапису	Аудіозапис трансформовано/не трансформовано у текст.
Автопунктуація	Розстановка знаків пунктуації у реченнях відповідно до змісту.
2. Користувачі системи	
Користувач	Будь-яка людина, що користується додатком.
3. Вхідні та вихідні документи	
Аудіозапис	Звуковий ряд, що містить людську мову без сторонніх звуків.
Текстове представлення	Текстовий документ, що містить текст відповідний його аудіозапису з пунктуацією.

2.2 Розроблення варіантів використання

У цьому підрозділі здійснюється формалізація функціональних вимог до системи шляхом побудови варіантів використання (use cases). Варіанти використання відображають основні сценарії взаємодії користувачів із системою, що дозволяє чітко визначити її функціональні межі та поведінку в типових ситуаціях.

2.2.1 Діаграма варіантів використання

Для проектування інформаційних систем широко використовується мова UML – уніфікована мова візуального моделювання [38]. UML може бути застосовано на всіх етапах життєвого циклу аналізу бізнес-систем і розробки прикладних програм. Різні види діаграм які підтримуються UML, і найбагатший набір можливостей представлення певних аспектів системи робить UML універсальним засобом опису як програмних, так і ділових систем.

Діаграма варіантів використання показана на рисунку (рисунок 2.1), призначена для графічного зображення основних функцій розроблюваного додатку та виявлення зв'язків та послідовностей варіантів використання.

У ході проектування діаграми використання були виявлені наступні основні варіанти використання.

Реєстрація/Вхід. Користувач додатку може увійти, або, у разі відсутності аккаунту, зареєструватися у системі, для того щоб мати доступ до створених матеріалів на інших пристроях.

Запис та збереження аудіозаписів. Користувач має можливість створення нового аудіозапису для подальшого використання та ввести його опис.

Перегляд списку аудіозаписів. Користувач може переглянути список своїх аудіозаписів та їх статусів.

Вибір та прослуховування аудіозапису. Користувач може обрати аудіозапис зі списку та прослухати його.

Видалення аудіозапису.

Перегляд інформації про аудіозапис. Користувач може переглянути опис аудіозапису та текстове його представлення у разі його наявності.

Трансформування аудіозапису у текст.

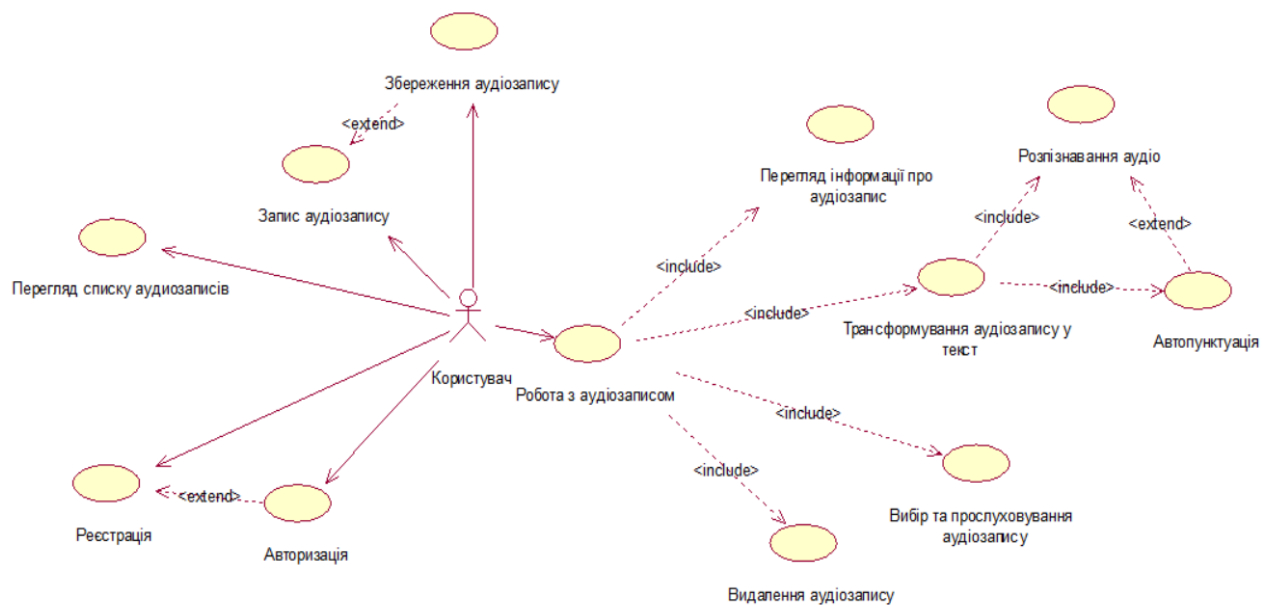


Рисунок 2.1 – Діаграма варіантів використання

2.2.2 Специфікація варіантів використання

В даному підрозділі описуються варіанти використання показані на рисунку (рисунок 2.1). У таблицях (таблиця 2.2 – 2.9) описані основні варіанти використання. Нижче представлені такі варіанти використання як: реєстрація у системі, авторизація у системі, запис та збереження аудіозапису, вибір та прослуховування аудіозапису, видалення аудіозапису, перегляд інформації про аудіозапис, трансформування аудіозапису у текст. Ці варіанти використання забезпечують основну функціональність розроблюваного додатку.

Таблиця 2.2 – Варіант використання «Реєстрація у системі»

Контекст використання	ЛТ-01 Реєстрація у системі
Дійові особи	Користувач
Предумова	
Триггер	Користувач бажає зареєструватися
Сценарій	<ol style="list-style-type: none"> 1. Зайти у додаток 2. Ввести логін, пароль та його підтвердження 3. Обрати «Зареєструватися»
Постумова	Якщо варіант використання виконався успішно, то користувач потрапить у головне меню додатку. Інакше якщо користувач з таким логіном вже існує, або введені паролі не співпадають, то з'явиться відповідне повідомлення і можна буде спробувати знову.

Таблиця 2.3 – Варіант використання «Авторизація у системі»

Контекст використання	ЛТ-02 Авторизація у системі
Дійові особи	Користувач
Предумова	Користувач зареєстрований у системі.
Триггер	Користувач бажає авторизуватися у системі
Сценарій	<ol style="list-style-type: none"> 1. Зайти у додаток 2. Ввести логін та пароль 3. Обрати «Увійти»
Постумова	Якщо варіант використання виконався успішно, то користувач потрапить у головне меню додатку. Інакше з'явиться повідомлення, що логін або пароль невірні і можна буде спробувати знову.

Таблиця 2.4 – Варіант використання «Запис аудіозапису»

Контекст використання	LT-03 Запис аудіозапису
Дійові особи	Користувач
Предумова	Користувач авторизований у системі.
Триггер	Користувач бажає записати аудіозапис
Сценарій	<ol style="list-style-type: none"> 1. Обрати «+» у верхньому меню 2. Обрати іконку мікрофону 3. Надати права доступу до мікрофону 4. Вимовити необхідний текст для запису 5. Обрати іконку зупинки
Постумова	Якщо варіант використання виконався успішно, то запис буде створено та з'явиться вікно для підтвердження збереження. Інакше у разі помилки запису, або у разі відсутності прав доступу до мікрофону з'явиться відповідне повідомлення.

Таблиця 2.5 – Варіант використання «Збереження аудіозапису»

Контекст використання	LT-04 Збереження аудіозапису
Дійові особи	Користувач
Предумова	Користувач авторизований у системі та записав новий аудіозапис.
Триггер	Користувач бажає зберегти аудіозапис
Сценарій	<ol style="list-style-type: none"> 1. Заповнити ім'я аудіозапису 2. Заповнити опис за вибором 3. Обрати чи потрібно трансформувати аудіозапис одразу 4. Натиснути Save

Продовження таблиці 2.5

Постумова	Якщо варіант використання виконався успішно, то запис буде збережено. Інакше у разі помилки створення або збереження файлу з'явиться відповідне повідомлення.
-----------	---

Таблиця 2.6 – Варіант використання «Вибір та прослуховування аудіозапису»

Контекст використання	ЛТ-05 Вибір та прослуховування аудіозапису
Дійові особи	Користувач
Предумова	Користувач авторизований у системі. Наявне підключення до Інтернету.
Триггер	Користувач бажає прослухати аудіозапис
Сценарій	1. Обрати необхідний аудіозапис зі списку 2. Обрати іконку відтворення 3. У разі необхідності зупинки обрати іконку паузи
Постумова	Якщо варіант використання виконався успішно, то обраний аудіозапис буде відтворено. Інакше, якщо файл зіпсовано або відсутній, або у разі відсутності інтернету для його скачування, з'явиться відповідне повідомлення.

Таблиця 2.7 – Варіант використання «Видалення аудіозапису»

Контекст використання	ЛТ-06 Видалення аудіозапису
Дійові особи	Користувач
Предумова	Користувач авторизований у системі та має аудіозапис.
Триггер	Користувач бажає видалити аудіозапис

Продовження таблиці 2.7

Сценарій	<ol style="list-style-type: none"> 1. Обрати необхідний аудіозапис зі списку 2. Провести ліворуч 3. Обрати іконку корзини 4. Підтвердити видалення
Постумова	Якщо варіант використання виконався успішно, то обраний аудіозапис буде видалено. Інакше з'явиться повідомлення і стан системи не зміниться.

Таблиця 2.8 – Варіант використання «Перегляд інформації про аудіозапис»

Контекст використання	LT-07 Перегляд інформації про аудіозапис
Дійові особи	Користувач
Предумова	Користувач авторизований у системі та має створений аудіозапис. Наявне підключення до Інтернету.
Триггер	Користувач бажає переглянути інформацію про аудіозапис
Сценарій	1. Обрати необхідний аудіозапис зі списку
Постумова	Якщо варіант використання виконався успішно, то з'явиться представлення з інформацією аудіозапису. Інакше з'явиться повідомлення і стан системи не зміниться.

Таблиця 2.9 – Варіант використання «Трансформування аудіозапису у текст»

Контекст використання	LT-08 Трансформування аудіозапису у текст
Дійові особи	Користувач
Предумова	Користувач авторизований у системі. Наявне підключення до Інтернету.

Продовження таблиці 2.9

Триггер	Користувач бажає трансформувати аудіозапис у текст
Сценарій	1. Обрати необхідний аудіозапис зі списку 2. Обрати опцію трансформування аудіозапису
Постумова	З'явиться спінер, що свідчить про те, що система обробляє запит. Якщо варіант використання виконався успішно, то аудіозапис буде трансформовано у текст з пунктуацією. Інакше стан системи не зміниться і з'явиться відповідне повідомлення з помилкою.

2.3. Специфікація функціональних та нефункціональних вимог

Функціональні вимоги – це опис того, що саме повинна виконувати система для задоволення потреб користувачів і досягнення бізнес-цілей. Вони охоплюють конкретну функціональність програмного забезпечення: доступні користувачам операції, обробку даних, логіку роботи системи та взаємодію з іншими компонентами.

Специфікація функціональних та нефункціональних вимог представлена у таблицях (таблиця 2.10, 2.11).

Таблиця 2.10 – Специфікація функціональних вимог

Ідентифікатор вимоги	Назва вимоги	Атрибути вимог	
		Пріоритет	Трудність
1	2	3	4
LT-01	Реєстрація у системі	рекомендоване	середня
LT-02	Авторизація у системі	рекомендоване	середня
LT-03	Запис аудіозапису	обов'язкове	середня

Продовження таблиці 2.10

1	2	3	4
LT-04	Збереження аудіозапису	обов'язкове	середня
LT-05	Вибір та прослуховування аудіозапису	рекомендоване	Низька
LT-06	Видалення аудіозапису	рекомендоване	низька
LT-07	Перегляд інформації про аудіозапис	обов'язкове	низька
LT-08	Трансформування аудіозапису у текст	обов'язкове	висока

Таблиця 2.11 – Специфікація нефункціональних вимог

Ідентифікатор вимоги	Назва вимоги	Атрибути вимог	
		Пріоритет	Трудність
1	2	3	4
1. Застосовність			
LT-09	Відповідність стандартам Apple, щодо графічного інтерфейсу	рекомендовано	середня
2. Надійність			
LT-10	Точність автопунктуації – не більше п'яти помилок на сто слів.	рекомендовано	висока
3. Робочі характеристики			
LT-11	Використання потоків для підвищення продуктивності	обов'язково	низька
4. Експлуатаційна придатність			
LT-12	Стандарти кодування - SwiftLint	рекомендовано	низька

Продовження таблиці 2.11

1	2	3	4
5. Проектні обмеження			
LT-13	Мова програмування – Swift	обов'язково	середня
6. Вимоги до документації, призначеної для користувача, і до системи допомоги			
LT-14	Видалення, відмова від збереження даних та вихід з аккаунту мають супроводжу- ватися додатковими повідомленнями	обов'язково	низька
7. Куповані компоненти			
LT-15	Використання Firebase	обов'язково	середня
LT-16	Використання Speechmatics	обов'язково	висока
8. Інтерфейси			
LT-17	Взаємодія з мікрофоном девайсу	обов'язково	низька
LT-18	Доступ до файлів девайсу	обов'язково	низька

2.4. Проектування інтерфейсу користувача

Інтерфейс користувача забезпечує зручну взаємодію із системою та дозволяє швидко виконувати ключові дії. Для демонстрації основної логіки роботи було створено мокапи основних екранів веб-застосунку які зображені на рисунках (рисунок 2.2 – 2.6).

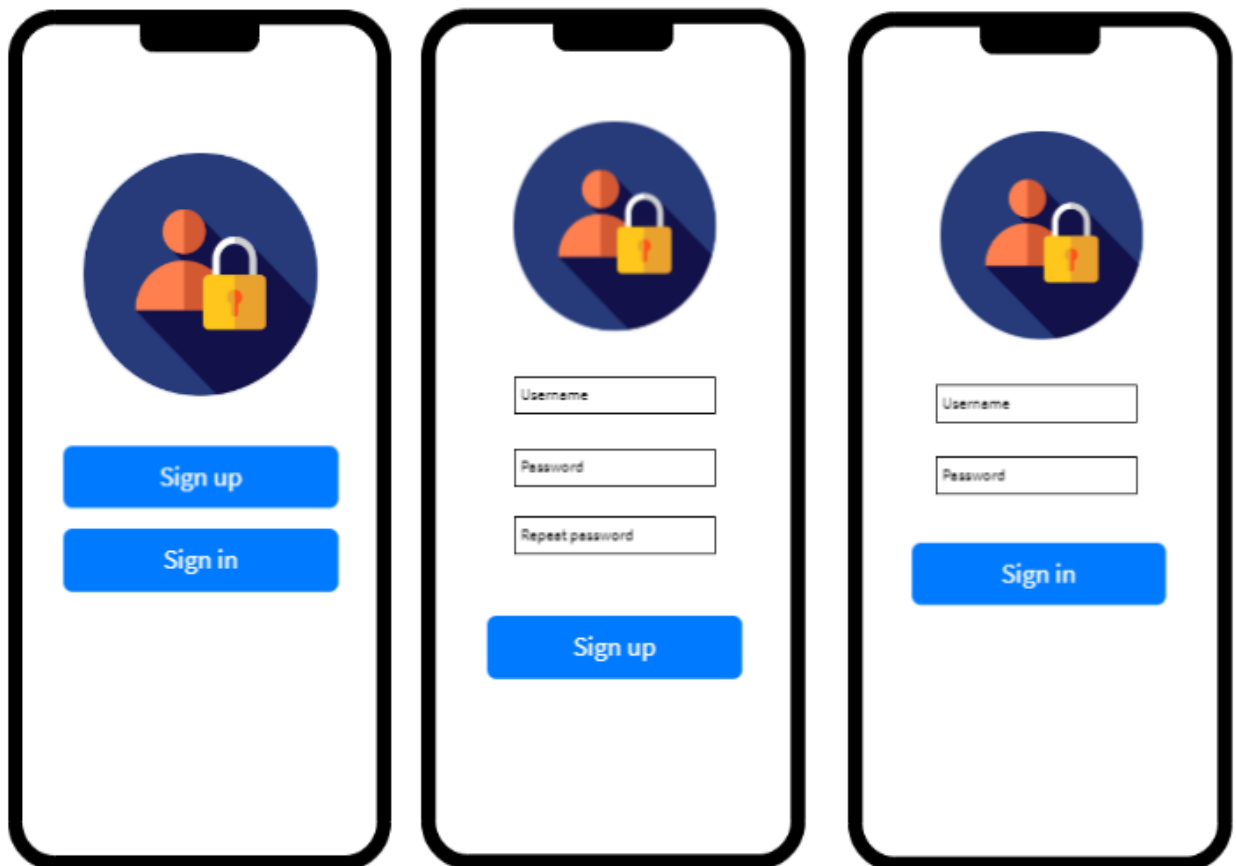


Рисунок 2.2 – Мокап авторизації

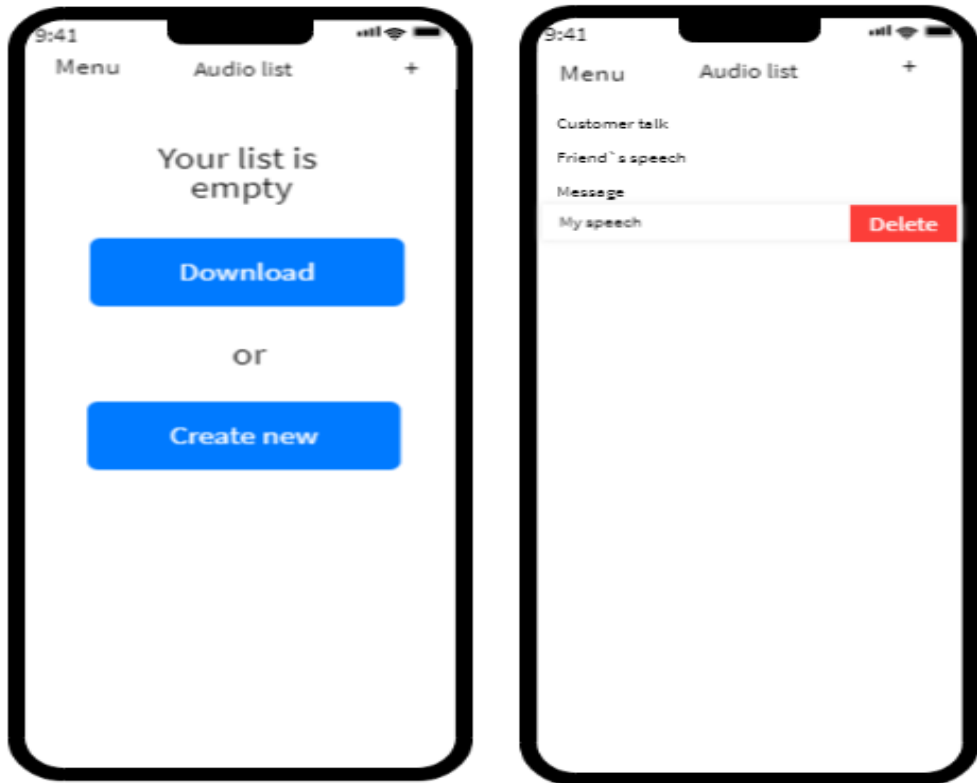


Рисунок 2.3 – Мокап списку аудіозаписів

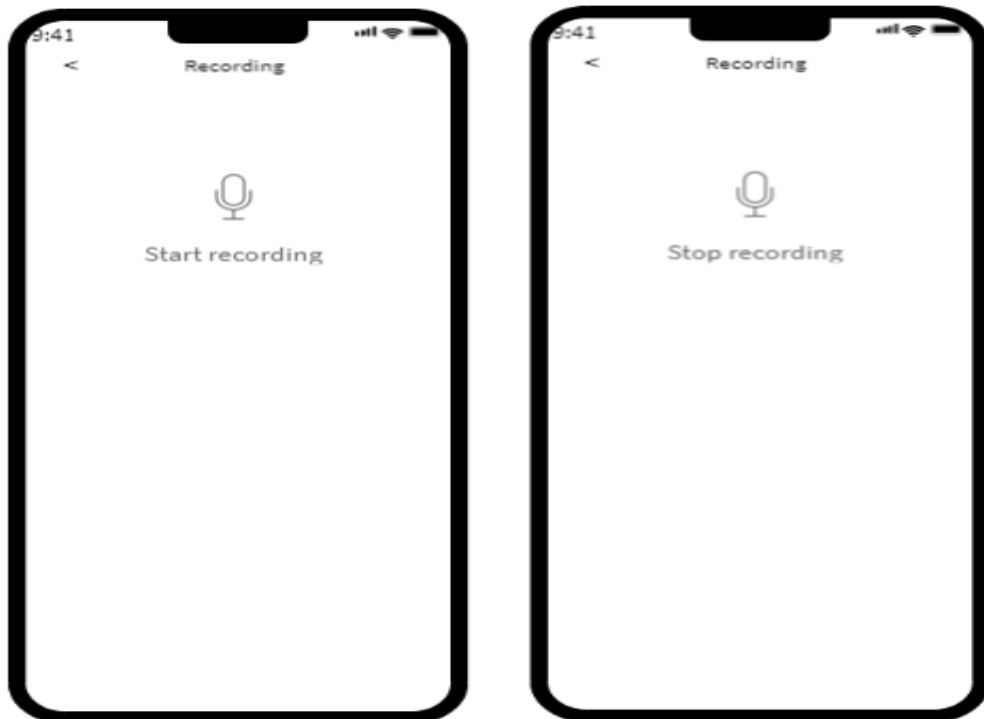


Рисунок 2.4 – Мокап запису нового аудіо ісів



Рисунок 2.5 – Мокап збереження нового аудіо



Рисунок 2.6 – Мокап перегляду інформації щодо аудіо

3. ПРОЄКТНІ ТА ТЕХНІЧНІ РІШЕННЯ

3.1. Логічна постановка

Основними логічними задачами є: запис аудіозапису, задача перетворення його у текст та задача автопунктуації отриманого тексту. Ці задачі буде реалізовано для англійської мови.

Для цього буде реалізовано UI та backend частини.

UI частина передбачає інтерфейс створення та роботи з аудіозаписом.

Backend частина передбачає відправку запиту на розпізнавання аудіозапису та використання моделі автопунктуації.

У першу чергу вирішується задача розпізнавання, а потім використовуючи отриманий текст, вирішується задача автопунктуації.

Серед наявних методів розпізнавання тексту є такі як:

- нативна бібліотека Speech, що дозволяє розпізнавати текст з високою точністю та без інтернету, що є більш безпечним, але вона дозволяє записувати звук у розмірі хвилина за сесію та має ліміт на хвилини, що доступні для використання;

- Snowboy – більш підходить для розпізнавання ключового слова, яке має бути триггером для подальших дій, має невеликий список натренованих моделей ключових слів;

- OpenEars – опенсорсна бібліотека, що також більш підходить для розпізнавання ключового слова, але має меншу точність і потребує створення власного словника;

- Google speech Api – потребує інтернету, але має можливість безперервного розпізнавання потокового аудіо, наявно багато мов для розпізнавання, передбачена конфіденційність даних. Також є можливість вибору між різними моделями машинного навчання залежно від того, для чого використовується програма, має безкоштовну версію, але потребує дані

кредитної картки для початку роботи;

- Speechmatics Api – підтримує багато аудіоформатів та мов, є доволі точною та може використовуватися для офлайн-обробки файлів, є платною, але надає 60 безкоштовних хвилин [36].

Отже, задача перетворення мови у текст буде вирішена за допомогою Speechmatics API. Вона розпізнає текст та поділяє його на смислові сегменти, а також визначає ким із співрозмовників була сказана та чи інша репліка.

Задача автопунктуації буде вирішена шляхом використання API, що використовує модель BERT [3]. BERT це метод обробки природної мови, заснований на використанні нейромереж нової архітектури для роботи з послідовностями, відомих як «трансформери». Ця технологія допомагає Google краще визначати контекст слів в пошукових запитах. Проривним моментом в роботі BERT є його здатність навчати мовні моделі на основі всього набору слів в додатку або запиті (двонаправленне навчання), тоді як в традиційному навчанні аналізується впорядкована послідовність слів (зліва направо або справа наліво). BERT дозволяє мовній моделі розуміти контекст слова на основі оточуючих її слів, а не тільки того слова, яке йому передуює або слід відразу за ним. Google називає BERT «глибоко двонаправленим», оскільки контекстні уявлення слів починаються «з самого низу глибокої нейронної мережі».

Іншою альтернативою є використання моделей RNN, які є також популярними у вирішенні задач природної обробки мови. Ідея RNN полягає в послідовному використанні інформації [14]. RNN називаються рекурентними, тому що вони виконують одну й ту ж задачу для кожного елемента послідовності, причому вихід залежить від попередніх обчислень. Найбільш часто використовуваним типом RNN є LSTM, які набагато краще зберігають довгострокові залежності, ніж RNN.

3.2. Проектування структури бази даних

3.2.1. Концептуальне інфологічне проектування

Інфологічне проектування – побудова семантичної моделі предметної області, тобто інформаційної моделі найбільш високого рівня абстракції [13]. Така модель створюється без орієнтації на якусь конкретну СУБД і модель даних. Інфологічне проектування бази передбачає створення словника даних наведеного нижче у таблиці (таблиця 3.1).

Таблиця 3.1 – Словник даних

№ з/п	Найменування елемента	Тип і довжина	Призначення елемента
1	audioname	String	назва аудіозапису
2	container FileName	String	базове ім'я файлу, що містить аудіозапис та його текстове представлення
3	dateRecorded	Date	дата створення аудіозапису
4	description	String	опис аудіозапису
6	is Transformed	Bool	інформація про те чи був аудіозапис трансформований
7	recordings	[Audio]	масив аудіозаписів користувача
8	username	String	ім'я користувача
9	transformed Text	String	розпізнаний текст з автопунктуацією

Отже, виходячи з таблиці 3.1 можна виявити дві сутності User та Audio (таблиця 3.2).

Таблиця 3.2 – Обмеження атрибутів сутностей

№ з/п	Ім'я атрибута або агрегату	Допустимі значення	Формат	Значення за замовчуванням
1	audioname	not null	String	“”
2	container File Name	not null	String	дата+час+“file”
3	date Recorded	not null	Date	поточна дата
4	description		String	“”
5	format	not null	String	“mp4”
6	ss Transformed	not null	Bool	false
7	recordings		[Audio]	[]
8	username	not null	String	“”
9	transformed Text		String	“”

3.2.2 Проектування логічної моделі даних

Логічне проектування – створення схеми бази даних на основі конкретної моделі даних, наприклад, реляційної моделі БД [13]. Для реляційної моделі даних – це набір схем відносин, зазвичай із зазначенням первинних ключів, а також «зв'язків» між відносинами, що представляють собою зовнішні ключі.

Зв'язок між виявленими сутностями User та Audio – один до багатьох (рисунок 3.1).

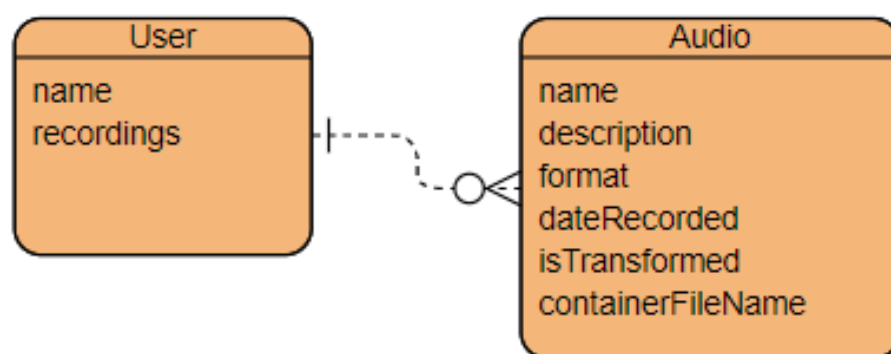


Рисунок 3.1 – ER-діаграма

У якості СУБД було обрано Firebase, тому що, окрім його використання у якості бази даних, для збереження матеріалів користувача буде використано сховище Firebase. Також Firebase має готові рішення для реалізації автентифікації [29].

Незважаючи на деякі недоліки, Firebase є гарним рішенням для системи з невеликою базою даних.

3.2.3. Проектування фізичної моделі даних

Фізичне проектування – створення схеми бази даних для конкретної СУБД [13]. Специфіка конкретної СУБД може включати в себе обмеження на іменування об'єктів бази даних, обмеження на підтримувані типи даних та інші. Крім того, специфіка конкретної СУБД при фізичному проектуванні включає вибір рішень, пов'язаних з фізичним середовищем зберігання даних (вибір методів управління дисковою пам'яттю, поділ БД по файлам і пристроям, методів доступу до даних), створення індексів та інші.

Firestore має два типи баз даних. Для рішення даного завдання було обрано базу Cloud-based, тому що вона є новою, тому більш швидкою та гнучкою, з урахуванням недоліків real-time, а саме: була покращена модель зберігання даних, додано можливість offline підтримки WEB застосунків, збільшена продуктивність запитів [28].

3.3. Проектування програмного забезпечення

Діаграма класів служить для подання статичної структури моделі системи в термінології класів об'єктно-орієнтованого програмування [12]. Діаграма класів може відбивати, зокрема, різні взаємозв'язки між окремими сутностями предметної області, такими як об'єкти й підсистеми, а також описує їхню внутрішню структуру й типи відносин.

Нижче, на рисунку (рисунок 3.2), показана безпосередньо діаграма

класів до даного додатку.

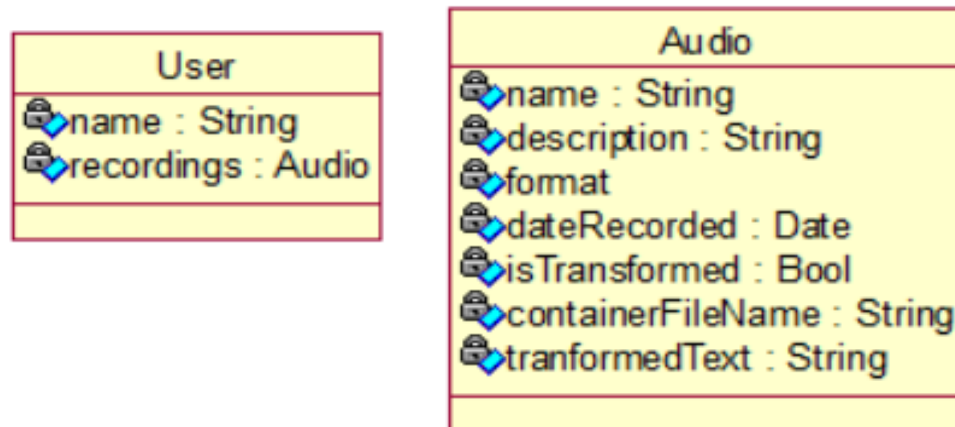


Рисунок 3.2 – Діаграма класів

У класі User зберігається інформація користувача, а саме: ім'я та масив аудіозаписів. Аудіозаписи представлені класом Audio, що містить такі властивості як: ім'я, опис, формат, дату створення, інформацію про те, чи був він трансформований, результат трансформації та ім'я файлу у якому зберігається аудіозапис.

Діаграма послідовності – різновид діаграми в UML [5]. Діаграма послідовності відображає взаємодії об'єктів впорядкованих за часом. Зокрема, такі діаграми відображають задіяні об'єкти та послідовність відправлених повідомлень.

Нижче, на рисунку (рисунок 3.3), показана безпосередньо діаграма послідовності до даного додатку.

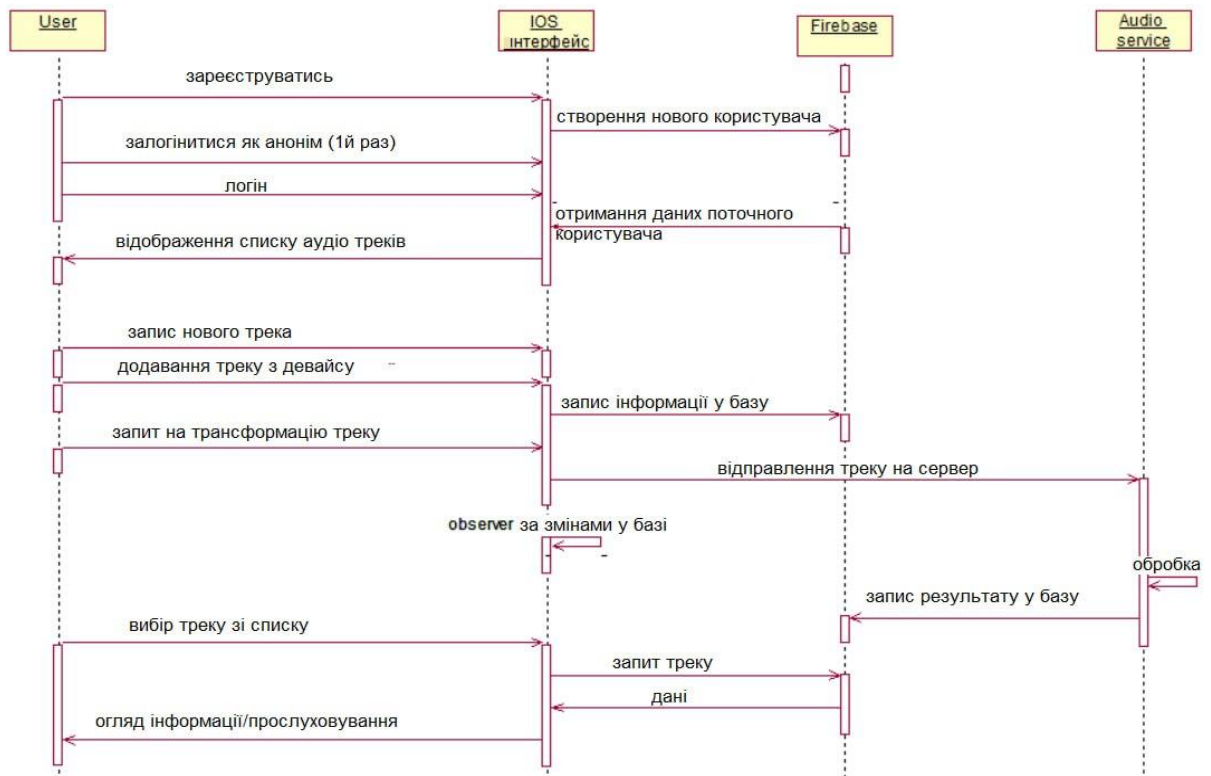


Рисунок 3.3 – Діаграма послідовності

За допомогою діаграми послідовності можна зручно простежити взаємодію користувача з мобільним додатком, а мобільного додатка – з базою даних та сервісом обробки мови.

Діаграма станів взаємно пов'язує події та стани. У разі приймання події наступний стан системи залежить як від поточного стану, так і від події. Зміна стану називається переходом.

Нижче, на рисунку (рисунок 3.4), показана безпосередньо діаграма станів до даного додатку.

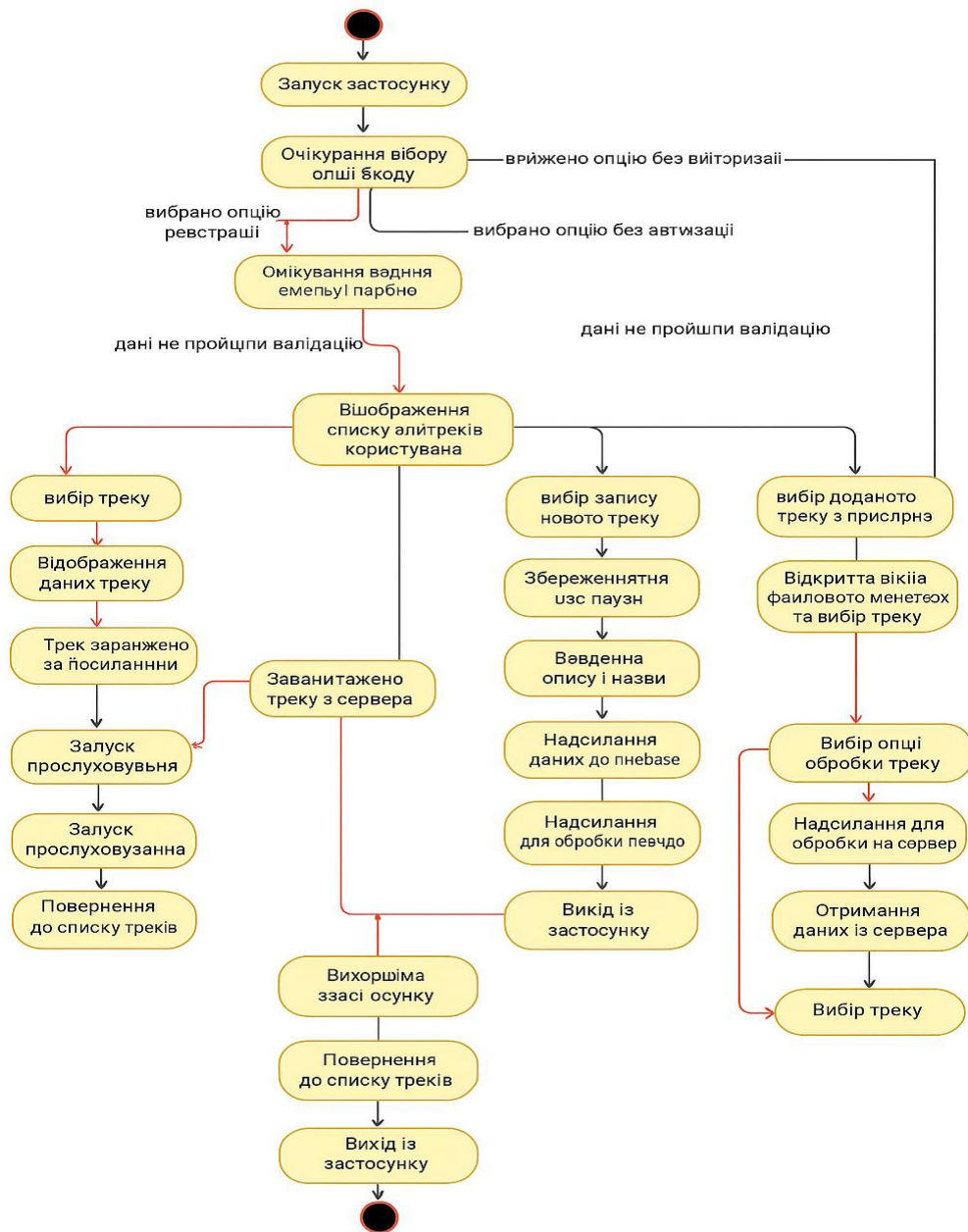


Рисунок 3.4 – Дїаграма станїв

Система має мїкросервїсну архїтектуру, тобто UI частина взаємодїє з бекенд частиною та базою даних.

Бекенд частина представлена через Speechmatics API та API, що використовує модель BERT, а базою даних є Firebase. Взаємодїя цих сервїсів показана на рисунку (рисунок 3.5). Лїстинг програми щодо основних бїзнес-процесїв представлений у додатку А.

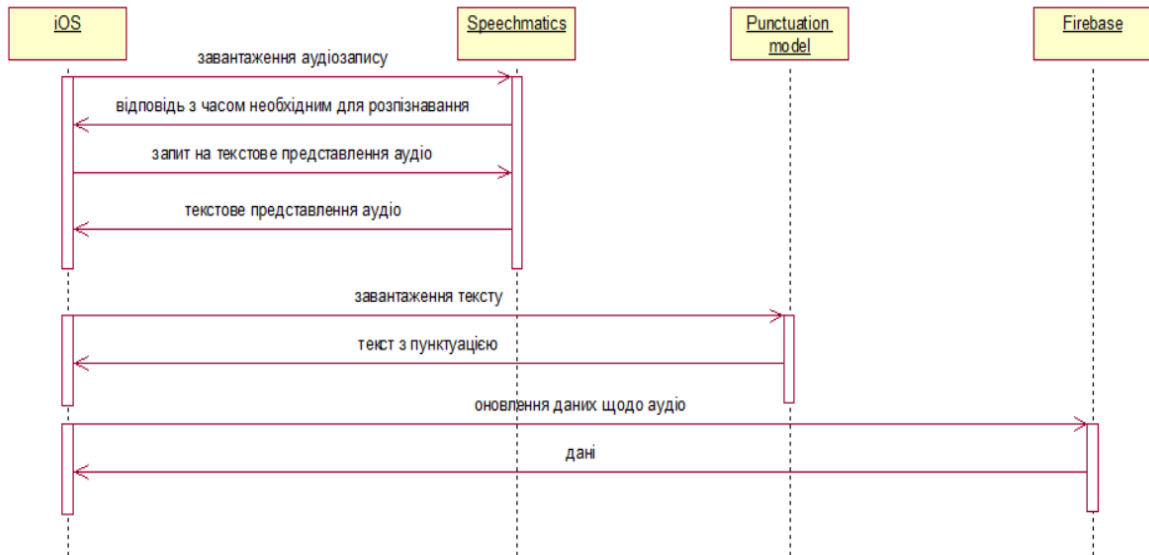


Рисунок 3.5 – Діаграма взаємодії сервісів

При проектуванні інтерфейсу користувача було передбачено: виведення інформаційних повідомлень у разі некоректних дій або даних, відображення стану системи за допомогою спінерів, використання елементів звичних користувачеві, попередження щодо деструктивних дій (рисунки 3.6 – 3.11).

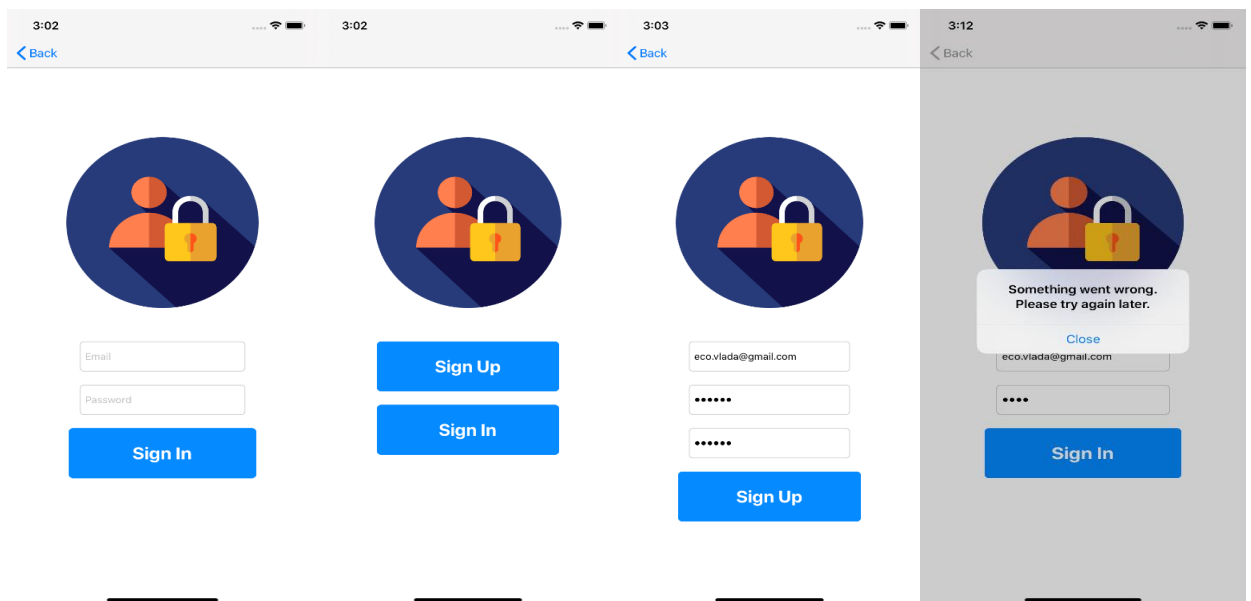


Рисунок 3.6 – Інтерфейс авторизації та взаємодія з ним

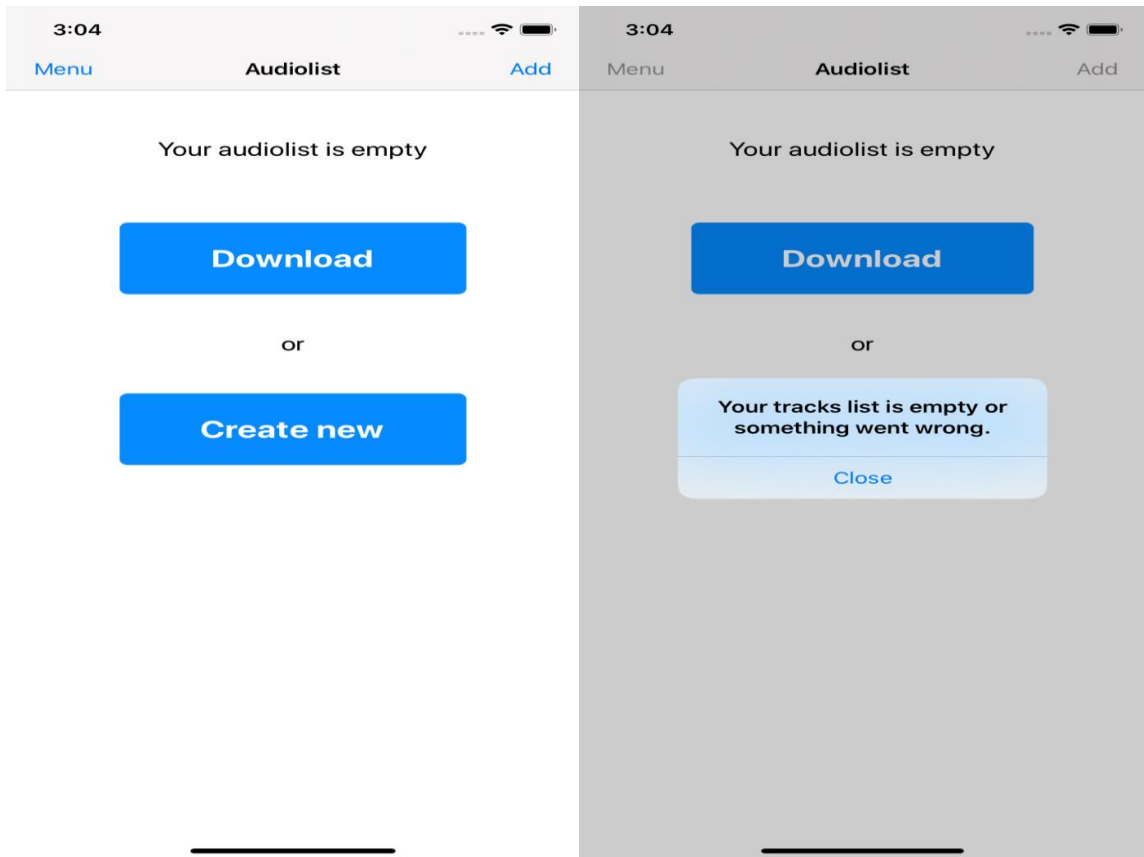


Рисунок 3.7 – Інтерфейс порожнього списку аудіозаписів

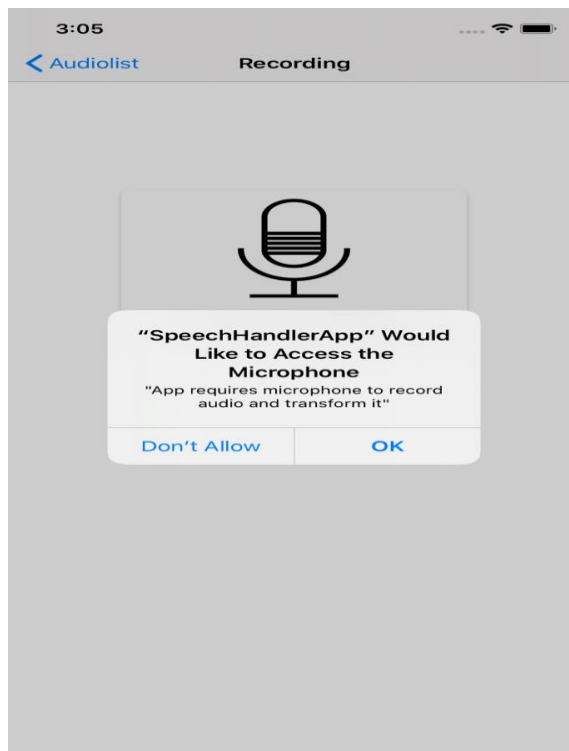


Рисунок 3.8 – Інтерфейс запису аудіо

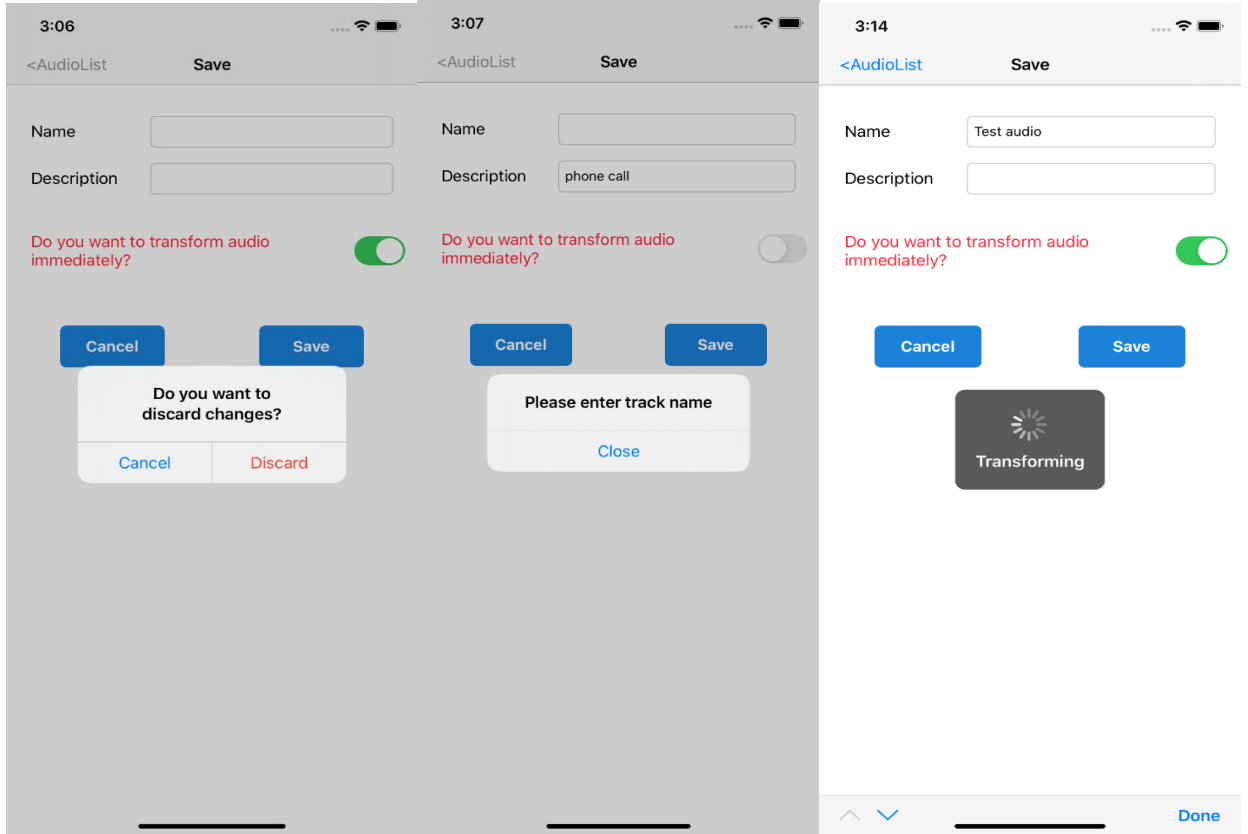


Рисунок 3.9 – Інтерфейс збереження нового аудіозапису та взаємодії з ним

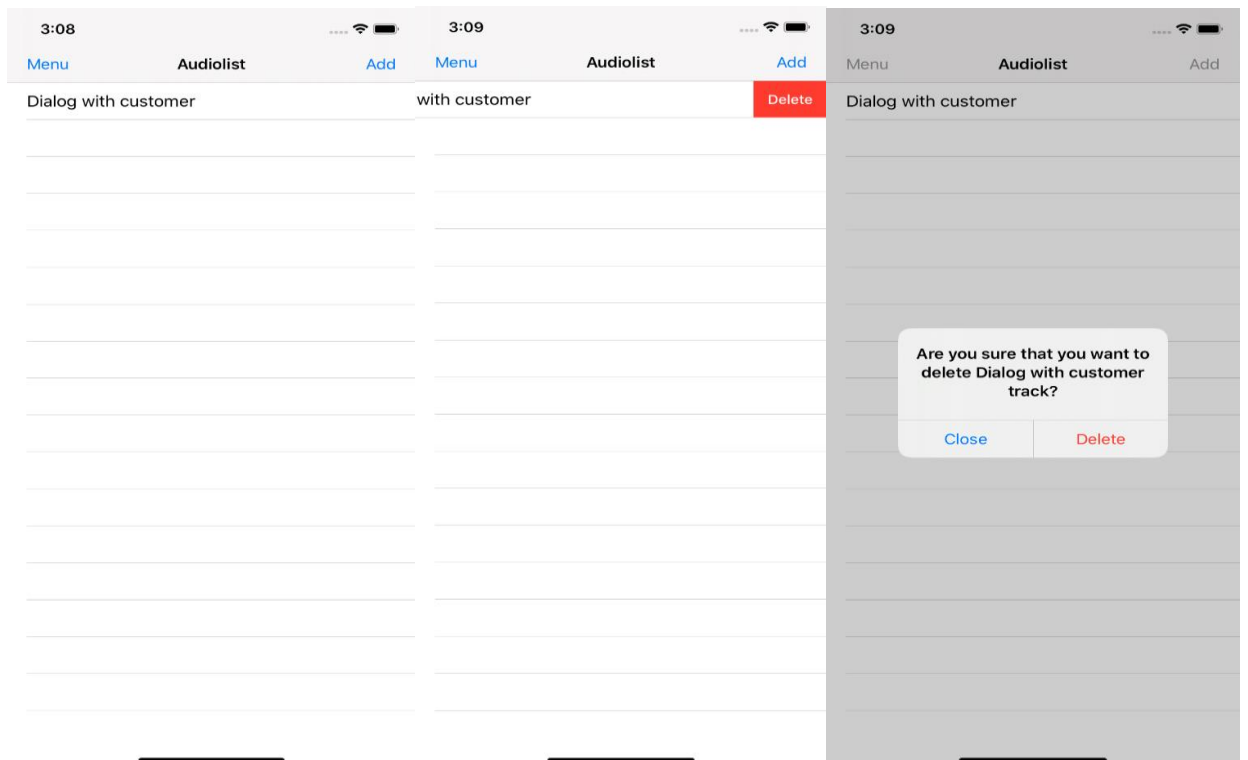


Рисунок 3.10 – Інтерфейс списку аудіозаписів та взаємодії з ним

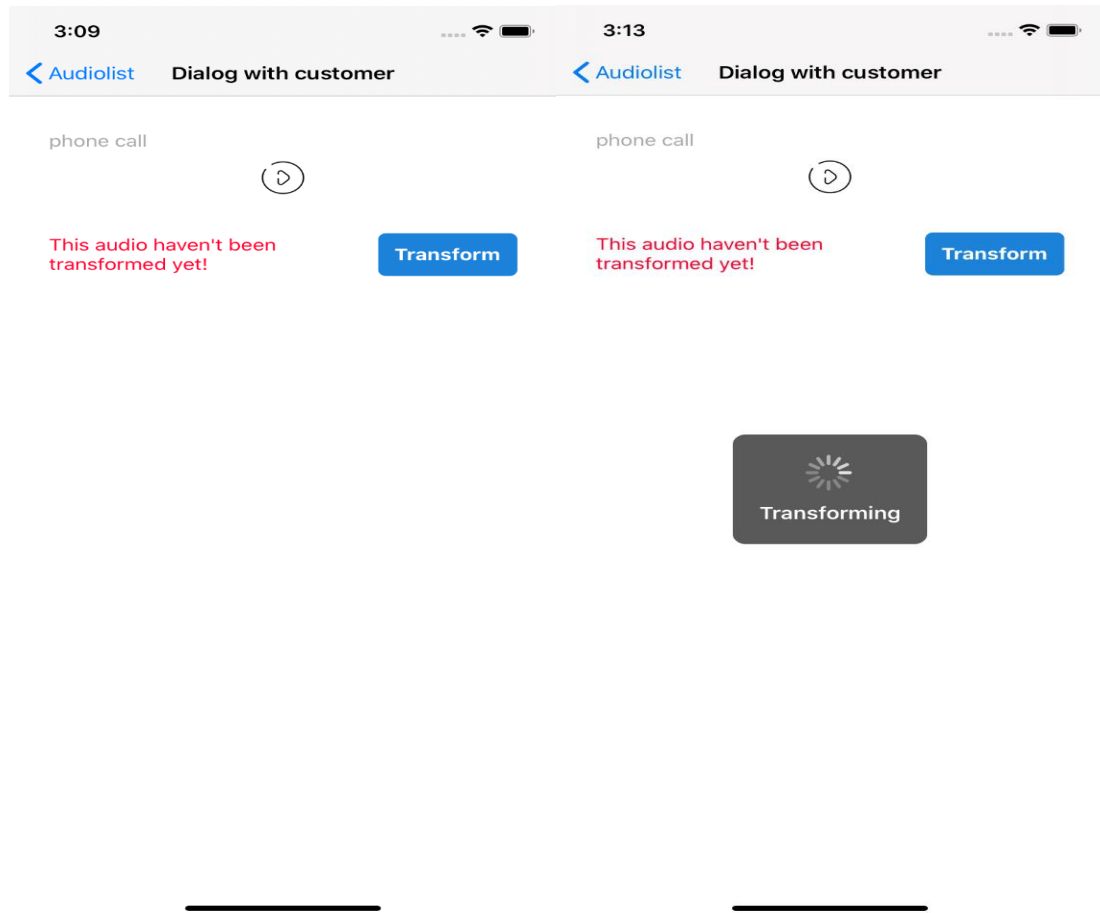


Рисунок 3.11 – Інтерфейс редагування наявного аудіозапису

3.4. Тестування програмної системи

Тест-вимоги, щодо реєстрації у системі (таблиця 3.3, 3.4).

При реєстрації у системі користувач має ввести емейл, пароль та його підтвердження. У разі якщо емейл є невалідним або паролі не співпадають, або пароль містить менше п'яти символів, користувач отримає відповідне повідомлення. Якщо дані вірні, то буде створено новий обліковий запис та користувач увійде у систему. Поля для заповнення мають містити підказки.

Таблиця 3.3 – Тест-кейс для тестування реєстрації

Дія	Очікуваний результат	Отриманий результат
Обрати “Sign Up”	Відкрито вікно реєстрації	+
Ввести: емейл test@test.com пароль test12 у поле пароля та поле його повторення	Пароль перетворюється на крапки	+
Обрати “Sign Up”	Відкрито головне вікно додатку	+
Результат: пройдено		

Таблиця 3.4 – Тест-кейс для тестування реєстрації

Дія	Очікуваний результат	Отриманий результат
Обрати “Sign Up”	Відкрито вікно реєстрації	+
Ввести: емейл test@test.com пароль test у поле пароля та поле його повторення та обрати “Sign Up”	З’являться пові- домлення “Entered data is not valid”	+
Ввести: емейл test пароль test у поле пароля та поле його повторення та обрати “Sign Up”	З’являться пові- домлення “Entered data is not valid”	+
Ввести: емейл test@test.com пароль test у поле пароля та test11 у поле його повторення та обрати “Sign Up”	З’являться повідомлення “Entered data is not valid”	+
Результат: пройдено		

Тест-вимоги, щодо авторизації у системі (таблиця 3.5, 3.6)

При авторизації у системі користувач має ввести емейл та пароль. У

разі якщо такого користувача не існує, користувач отримає відповідне повідомлення. Якщо такий користувач існує, то він увійде у систему.

Таблиця 3.5 – Тест-кейс для тестування авторизації

Передумова: користувач зареєстрований у системі логін test@test.com, пароль test12		
Дія	Очікуваний результат	Отриманий результат
Обрати “Sign In”	Відкрито вікно авторизації	+
Ввести: емейл test@test.com пароль test12	Пароль перетворюється на крапки	+
Вимкнути інтернет та обрати “Sign In”	З’являться повідомлення “Something went wrong. Please try again later”	+
Ввімкнути інтернет та обрати “Sign In”	Відкрито головне вікно додатку	+
Результат: пройдено		

Таблиця 3.6 – Тест-кейс для тестування авторизації

Дія	Очікуваний результат	Отриманий результат
Обрати “Sign In”	Відкрито вікно авторизації	+
Ввести: емейл test@test.com пароль test та обрати “Sign In”	З’являться повідомлення “Something went wrong. Please try again later”	+
Результат: пройдено		

Тест-вимоги, щодо запису аудіозапису (таблиця 3.7, 3.8).

При створенні аудіопису користувач має отримати запит для доступу до мікрофону. У разі відмови користувача запис не буде створено та при

наступній спроби запису з’явиться відповідне повідомлення. У разі згоди про надання прав інтерфейс має змінитися для передачі стану запису.

Таблиця 3.7 – Тест-кейс для тестування запису аудіо

Передумова: користувач увійшов у систему та ще не створював аудіозаписи		
Дія	Очікуваний результат	Отриманий результат
Обрати “Create new” або “Add”	Відкрито вікно запису аудіо	+
Обрати “Start recording”	Запит на запис аудіо та доступ до мікрофону	+
Обрати “Don’t allow” у вікні запиту	Вікно закриється	+
Обрати “Start recording”	З’явиться повідомлення “App need record permissions for recording”. Please allow it and try again.	+
У налаштуваннях надати доступ до мікрофону, повернутися до додатку та обрати “Start recording”	“Start recording” зміниться на “Stop recording”	+
Обрати “Stop recording”	Відкрито вікно збереження аудіозапису	+
Результат: пройдений		

Таблиця 3.8 – Тест-кейс для тестування запису аудіо

Передумова: користувач увійшов у систему та ще не створював аудіозаписи		
Дія	Очікуваний результат	Отриманий результат
Обрати “Create new” або “Add”	Відкрито вікно запису аудіо	+

Продовження таблиці 3.8

Дія	Очікуваний результат	Отриманий результат
Обрати “Start recording”	Запит на запис аудіо та доступ до мікрофону	+
Обрати “Ok” у вікні запиту	Вікно закриється “Start recording” зміниться на “Stop recording”	+
Обрати “Stop recording”	Відкрито вікно збереження аудіозапису	+
Результат: пройдений		

Тест-вимоги, щодо збереження аудіозапису (таблиця 3.9 – 3.12).

При збереженні аудіозапису користувач має ввести його ім'я. У разі якщо користувач не бажає зберігати зміни має з'явитися попереджувальне повідомлення про втрату змін з можливістю відмінити цю дію.

Таблиця 3.9 – Тест-кейс для тестування збереження аудіозапису

Передумова: користувач записав аудіо		
Дія	Очікуваний результат	Отриманий результат
Обрати “<AudioList” або “Cancel”	З'явиться “Discard changes” вікно	+
Обрати “Cancel”	Вікно закриється	+
Обрати “AudioList”	Повернення до головного вікна, новий аудіозапис не створюється	+
Результат: пройдений		

Таблиця 3.10 – Тест-кейс для тестування збереження аудіозапису

Передумова: користувач записав аудіо		
Дія	Очікуваний результат	Отриманий результат
Ввести ім'я аудіозапису та вимкнути трансформування аудіо, обрати "Save"	Повернення до головного вікна, новий аудіозапис додано до списку	+
Результат: пройдено		

Таблиця 3.11 – Тест-кейс для тестування збереження аудіозапису

Передумова: користувач записав аудіо		
Дія	Очікуваний результат	Отриманий результат
Ввести ім'я аудіозапису, обрати "Save"	З'явиться спінер з написом "Transforming", повернення до головного вікна, новий аудіозапис додано до списку	+
Результат: пройдено		

Таблиця 3.12 – Тест-кейс для тестування збереження аудіозапису

Передумова: користувач записав аудіо		
Дія	Очікуваний результат	Отриманий результат
Ввести ім'я аудіозапису, вимкнути інтернет та обрати "Save"	З'явиться повідомлення "Audio can't be transformed now. Internet is turned off."	-
Результат: не пройдено		

Тест-вимоги, щодо прослуховування аудіозапису (таблиця 3.13).

При прослуховуванні аудіозапису у разі якщо система не має доступ до

файлу або не може його відтворити, то користувач отримує повідомлення.

Таблиця 3.13 – Тест-кейс для тестування прослуховування аудіозапису

Передумова: користувач має записане аудіо		
Дія	Очікуваний результат	Отриманий результат
Обрати аудіо зі списку	З'явиться вікно з інформацією щодо аудіо	+
Обрати ">"	Аудіо буде відтворено	+
Результат: пройдено		

Тест-вимоги, щодо видалення аудіозапису (таблиця 3.14).

При видаленні аудіозапису має з'явитися попереджувальне повідомлення про втрату змін з можливістю відмінити цю дію, інакше об'єкт має бути видалено.

Таблиця 3.14 – Тест-кейс для тестування видалення аудіозапису

Дія	Очікуваний результат	Отриманий результат
Потягнути рядок з аудіозаписом ліворуч	З'явиться кнопка "Delete"	+
Обрати "Delete"	З'явиться вікно "Are you sure that you want to delete speech track?"	+
Обрати "Close"	Вікно закриється	+
Обрати "Delete"	Аудіозапис буде видалений зі списку	+
Результат: пройдений		

Тест-вимоги, щодо трансформування аудіозапису у текст (таблиця 3.14, 3.15).

При трансформуванні аудіозапису система має передавати цей стан на

інтерфейсі. У разі помилок користувач отримає повідомлення.

Таблиця 3.15 – Тест-кейс для тестування трансформування аудіозапису у текст

Передумова: користувач має записане аудіо		
Дія	Очікуваний результат	Отриманий результат
Обрати аудіо зі списку	З'явиться вікно з інформацією щодо аудіо	+
Обрати "Transform"	З'явиться спінер з написом "Transforming", після його зникнення зникне секція трансформування та з'явиться відповідний трансформований текст	+
Результат: пройдено		

3.5. Розгортання програмного продукту

Вимоги до апаратних і програмних засобів: версія iOS 13+, ОЗУ 2+Гб.

Для поширення додатку на інші девайси, його має бути додано до App Store.

Для встановлення на девайсі необхідно буде знайти додаток у App Store та натиснути Install.

Кроки для додання додатку до AppStore [13]:

- реєстрація apple id;
- реєстрація облікового запису розробника у програмі розробників Apple;
- створення Bundle id додатку – com.radchenko.speechHadlerApp;
- створення сертифікату розробника.

Сертифікат являє собою пару ключів асиметричного шифрування:

приватний і публічний. У процесі збірки білду XCode формує цифровий підпис для збірки на підставі даних приватного ключа. Перевірити підпис можна за допомогою публічного ключа, який доступний і для Apple, що видала цей сертифікат. Одного сертифіката досить для завантаження необмеженої кількості додатків. Необхідно створити сертифікат типу Production – App Store and Ad Hoc, зберегти його та відкрити.

Створення AppId.

AppId це унікальний строковий ідентифікатор додатку. Він потрібен для однозначної ідентифікації в усіх системах: iTunes Connect, App Store та ін. Він складається з двох частин: TeamID і BundleID. TeamID - ідентифікатор розробника, видається Apple на етапі реєстрації аккаунта розробника і не змінюється. BundleID задається розробником при реєстрації додатку в обліковому записі.

Створення provisioning типу Production.

Це профіль, який однозначно пов'язує AppID і сертифікат розробника (публічний ключ сертифіката). У разі development provisioning profile він також містить UDID всіх пристроїв, на яких можливий запуск програми.

Реєстрація додатку у itunes.connect.

У Xcode встановити бажану версію, архівувати проект та обрати опцію Upload, проект з'явиться у AppStore.

Відправити версію на перевірку у Apple.

Після успішної перевірки додаток може бути додано до AppStore.

ВИСНОВКИ

У ході розробки проекту кваліфікаційної роботи було проаналізовано існуючі методи обробки мови, а також вивчена актуальність даної проблеми, також було змодельовано бізнес-процеси за допомогою IDEF0 та DFD.

Для графічного зображення основних функцій розроблюваного додатку, виявлення зв'язків та послідовностей варіантів використання було розроблено варіанти використання, а також спроектовано інтерфейс користувача.

Для програмної реалізації було спроектовано структуру бази даних, діаграму класів, станів та активності, а також змодельовано мікросервісну архітектуру додатку.

Отже, результатом розробки кваліфікаційної роботи є мобільний додаток з функціями розпізнавання та автопунктуації тексту.

Надалі необхідно покращувати якість розпізнавання та автопунктуації, а також можливе розширення функцій роботи з трансформованим текстом, а саме: надання можливостей редагування та збереження файлу з текстом, а також його розповсюдження через соціальні мережі.

Також необхідно надати можливість користування додатком без реєстрації та реєстрації використовуючи соціальні мережі.

ПЕРЕЛІК ДЖЕРЕЛ ПОСИЛАННЯ

1. Блокнот для мовного введення. [Електрон. ресурс]; Спосіб доступу url: <https://www.speechpad.com/>
2. Викладаємо додаток в App Store. Навіть якщо ви не розробник. [Електрон. ресурс]; Спосіб доступу url: <https://habr.com/ru/company/touchinstinct/blog/345336/>
3. Все, що потрібно знати про алгоритм BERT в пошуку Google [Електрон. ресурс]; Спосіб доступу url: <https://blogchain.com.ua/vse-shcho-potribno-znaty-pro-alhorytm-bert-v-poshuku-google/>
4. Голосовий набір тексту: онлайн-сервіси і додатки для комп'ютера. [Електрон. ресурс]; Спосіб доступу url: <https://habr.com/ru/companies/click/articles/534710/>
5. Діаграма послідовності. [Електрон. ресурс]; Спосіб доступу url: https://uk.wikipedia.org/wiki/Діаграма_послідовності
6. Діаграми потоків даних (DFD) як один з головних інструментів структурного аналізу та проектування інформаційних систем. [Електрон. ресурс]; Спосіб доступу url: <https://studopedia.org/8-191280.html>
7. Кафедра ЕОМ. [Електрон. ресурс]; Спосіб доступу url: <https://dec.nure.ua/uk-graduating/>
8. Методичні вказівки до організації виконання та захисту кваліфікаційних робіт першого (бакалаврського) рівня вищої освіти для здобувачів спеціальності 123 «Комп'ютерна інженерія» усіх форм навчання / Упорядники: О.С. Ляшенко, А.А. Коваленко Д.М. Росінський, І.В. Філіппенко, О.С. Шкіль. – Харків: ХНУРЕ, 2025. – 50 с.
9. Моделі потоків даних (DFD-моделі): призначення, місце застосування в системному аналізі, правила побудови, приклади. [Електрон. ресурс]; Спосіб доступу url: <https://studopedia.org/8-191272.html>
10. Потапов Р.О., Федорченко В.М. // Основні підходи до вирішення

завдань розпізнавання та автопунктуації тексту // Матеріали п'ятнадцятої міжнародної науково-технічної конференції “Сучасні напрями розвитку інформаційно-комунікаційних технологій та засобів управління” . Том 3 : тези доповідей, 24 – 25 квітня 2025 р. – Харків : ХНУРЕ, 2025. – С. 54.

11. Перші кроки в NLP: розглядаємо Python-бібліотеку TensorFlow та нейронні мережі в реальному завданні. [Електрон. ресурс]; Спосіб доступу url: <https://dou.ua/lenta/articles/first-steps-in-nlp-tensorflow/>

12. Призначення діаграми класів. [Електрон. ресурс]; Спосіб доступу url: https://life-prog.ru/ukr/1_4646_priznachennya-diagrami-klasiv.html

13. Проектування бази даних. [Електрон. ресурс]; Спосіб доступу url: https://uk.wikipedia.org/wiki/Проектування_бази_даних

14. Рекурентні нейронні мережі: типи, навчання, приклади та застосування. [Електрон. ресурс]; Спосіб доступу url: <https://itwiki.dev/data-science/ml-reference/ml-glossary/recurrent-neural-network>

15. Розроблення модуля для проведення вікторини на базі мобільних технологій. [Електрон. ресурс]; Спосіб доступу url: <https://studfile.net/preview/4643332/>

16. Штучна нейронна мережа [Електрон. ресурс]; Спосіб доступу url: https://uk.wikipedia.org/wiki/Штучна_нейронна_мережа

17. Що таке транскрибація аудіо або відео в тексті і як на цьому заробити? [Електрон. ресурс]; Спосіб доступу url: <https://www.grivnya.in.ua/yak-zarobiti-groshi/transkrybatsiia/>

18. Які аудіо формати підтримує iPhone, iPad, iPod? [Електрон. ресурс]; Спосіб доступу url: <https://vinnytsia.malevaro.cx.ua/articles/jakij-format-video-dlja-ajfona.html>

19. Як розмістити свій додаток в магазині App Store. [Електрон. ресурс]; Спосіб доступу url: <https://appbooster.com/knowledge/publikaciya-v-app-store/>

20. About TalkTyper. [Internet]; url: <https://talktyper.com/about.html>

21. Best Speech-to-Text APIs. [Internet]; url: <https://nordicapis.com/5-best->

[speech-to-text-apis/](#)

22. Comparison of Top 10 Speech Processing APIs. [Internet]; url: <https://medium.com/activewizards-machine-learning-company/comparison-of-top-10-speech-processing-apis-2293de1d337f>

23. Create a Storage Reference on iOS. [Internet]; url: <https://firebase.google.com/docs/storage/ios/create-reference>

24. DeepCorrection2: Automatic punctuation restoration. [Internet]; url: <https://medium.com/@praneethbedapudi/deepcorrection2-automatic-punctuation-restoration-ac4a837d92d9>

25. Delete data from Cloud Firestore. [Internet]; url: <https://firebase.google.com/docs/firestore/manage-data/delete-data>

26. Deploying a Deep Learning Model on Heroku using Flask and Python. [Internet]; url: <https://towardsdatascience.com/deploying-a-deep-learning-model-on-heroku-using-flask-and-python-769431335f66>

27. Deploying the machine learning model in Heroku using Flask. [Internet]; url: <https://becominghuman.ai/deploying-the-machine-learning-model-in-heroku-using-flask-549c9ad0a47a>

28. Get data with Cloud Firestore. [Internet]; url: <https://firebase.google.com/docs/firestore/query-data/get-data>

29. Get Started with Firebase Authentication on iOS. [Internet]; url: <https://firebase.google.com/docs/auth/ios/start>

30. How to Import Python Modules in Server Side Swift. [Internet]; url: <https://medium.com/server-side-swift-and-more/how-to-import-python-modules-in-server-side-swift-e50271c26853>

31. How to play sounds using AVAudioPlayer. [Internet]; url: <https://www.hackingwithswift.com/example-code/media/how-to-play-sounds-using-avaudioplayer>

32. How to record audio using AVAudioRecorder. [Internet]; url: <https://www.hackingwithswift.com/example-code/media/how-to-record-audio-using-avaudiorecorder>

33. IDEF0. [Internet]; url: <https://ru.wikipedia.org/wiki/IDEF0>
34. JGProgressHUD. [Internet]; url: <https://cocoapods.org/pods/JGProgressHUD>
35. Punctuator. [Internet]; url: <https://github.com/ottokart/punctuator2>
36. Speechmatics API Documentation. [Internet]; url: <https://app.speechmatics.com/api-details>
37. Steven Bird, Ewan Klein, and Edward Loper “Natural Language Processing with Python - Analyzing Text with the Natural Language Toolkit”. [Internet]; url: http://www.nltk.org/book_1ed/
38. Unified Modeling Language. [Internet]; url: https://uk.wikipedia.org/wiki/Unified_Modeling_Language
39. Upload Files on iOS. [Internet]; url: <https://firebase.google.com/docs/storage/ios/upload-files>
40. Voco. [Internet]; url: <https://www.speechpro.ru/product/programmy-dlya-raspoznavaniya-rechi-v-tekst/voco>
41. VoiceNote. [Internet]; url: <https://voicenote.in/live/>