

РОЗПОДІЛЕНЕ КОМП'ЮТЕРНЕ ДОКУМЕНТУВАННЯ ГОЛОСОВИХ МОВНИХ ФОНОГРАМ

Ю.Г. Кривонос, Ю.В. Крак, О.В. Бармак, О.С. Загваздін

Інститут кібернетики ім. В.М. Глушкова НАН України,
03680, МСП, Київ, проспект Академіка Глушкова, 40.
Тел.: (+38) (044) 526 2008, факс (+38) (044) 526 7418

Запропоновано підхід для розподіленого комп'ютерного документування мовних фонограм засідань. Проведено аналіз предметної області та зроблена постановка задачі. Побудована логічна модель інформаційної системи (ІС). Реалізований прототип ІС та окреслені необхідні дослідження для створення промислової версії ІС.

An approach to the distributed application speech phonogram is suggested. The project domain analysis is given. The information systems logical model is constructed. The information system example is given.

Якість інформаційного процесу підтримки діяльності представницьких органів, органів виконавчої влади, судів постійно знаходяться в центрі уваги, тому, що безпосередньо впливає на результати їх діяльності. Забезпечення оперативності, достовірності та багатосторонності поступаючої інформації дозволяє організувати цикл їх роботи з оптимальними часовими характеристиками.

Досвід показує, що простого нарощування обслуговуючого персоналу та технічних засобів недостатньо для вдосконалення управлінського процесу та переходу до керування інформацією. Для ситуацій, коли необхідна оперативна текстова розшифровка виступів, пропонується інформаційна система розподіленого комп'ютерного документування (ІС), яка дозволяє одночасно обробляти багато каналів поступаючої мовної інформації.

Принцип її роботи полягає у тому, що мовний сигнал, який поступає на вхід ІС (від конференц-системи або безпосередньо з мікрофону), записується на вінчестер комп'ютера, автоматично розбивається на окремі фрагменти та розподіляється між операторами-стенографістами, які прослуховують його та роблять текстову розшифровку. Отримані ділянки тексту автоматично з'єднуються в єдиний документ, який після перевірки керівником групи може бути збереженим та роздрукованим.

Міжнародний досвід використання таких систем показав їх високу ефективність. Слід відмітити подібні розробки «Центра речевих технологій» – російського лідера в галузі розробки електронної техніки та програмного забезпечення для високоякісного запису, обробки та аналізу звукової інформації [1], комплекс оперативного стенографування "SRS Report 2002" [2], розробки «International computer science institute» [3] тощо [4–7].

Авторами запропонована концепція та реалізація ІС «Автоматизована стенограма» для оперативного отримання стенограм виступів на базі технології розподіленої обробки мовного сигналу.

Аналіз предметної області та постановка задачі

Для автоматизованого ведення стенограм мають бути реалізовані основна функціональність та такі вимоги:

- мовний сигнал, який поступає із відповідної апаратури (мікрофони, звукові плати тощо):
 - а) записується на жорсткий диск комп'ютера у вигляді wav-файлу;
 - б) автоматично розбивається на окремі фрагменти заданої тривалості, які записуються в базу даних (БД) з відповідною ознакою (необроблений сегмент);
- адміністратор групи стенографістів у локальній мережі в реальному часі автоматично отримує інформацію про запис в БД нового необробленого сегмента;
- адміністратор групи стенографістів у реальному часі із локальної мережі отримує інформацію про статуси робочих місць операторів-стенографістів (оператор вільний чи обробляє сегмент);
- адміністратор, отримавши інформацію про необроблений сегмент, що поступив, підтверджує та направляє його вільному оператору-стенографісту (або ставить його в чергу для подальшої обробки);
- оператор-стенографіст:
 - 1) отримує необроблений фрагмент та, прослуховуючи його, – проводить текстову розшифровку;
 - 2) закінчивши розшифровку, записує фрагмент у БД;
- оброблені операторами-стенографістами сегменти тексту автоматично з'єднуються у підсумковий документ та інформація про це поступає адміністратору групи стенографістів;
- адміністратор групи стенографістів направляє підсумковий документ редактору-коректору;
- редактор-коректор, закінчивши роботу з підсумковим документом, направляє його адміністратору;

- адміністратор має змогу для поточного сигналу вказати прізвища людей, які приймають участь у засіданні та відповідні їм номери мікрофонів (які він попередньо отримує від організаторів засідання).

Далі, на основі цих вимог та обмежень, виділимо класи користувачів ІС, визначимо їх вимоги, та побудуємо опис системи з точки зору користувача. Складемо діаграму використання, яка описує ІС в цілому (рис. 1). Ця діаграма включає аспекти використання та зовнішні суб'єкти системи.

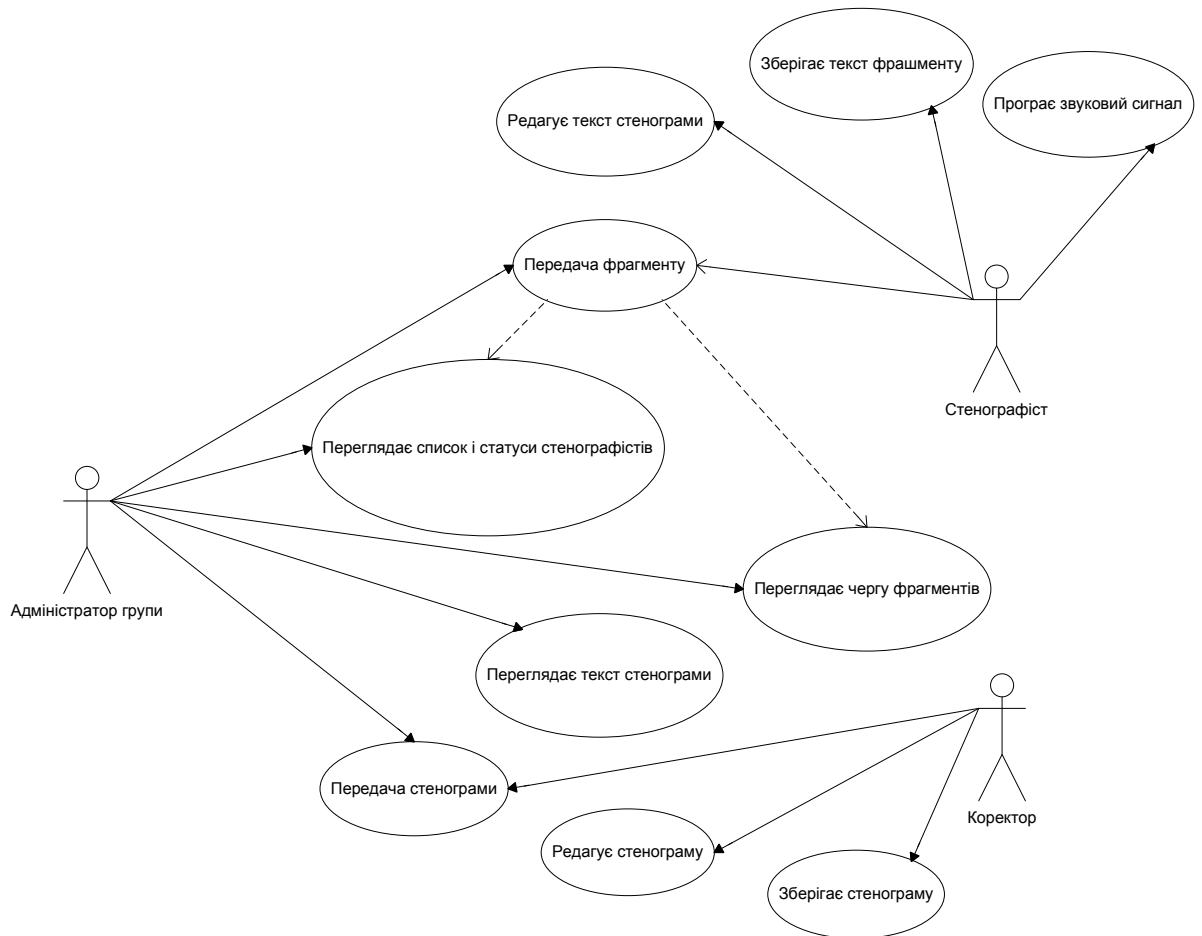


Рис. 1. Діаграма використання ІС автоматизованого ведення стенограм засідань

У ІС автоматизованого ведення стенограм засідань виділені три класи користувачів (три ролі), яким відповідають три зовнішні суб'єкти, представлених на діаграмі.

1. Адміністратор групи стенографістів – відповідає за розподіл завдань між операторами-стенографістами та редакторами-коректорами, контролює хід виконання завдань.

2. Оператор-стенографіст – виконує завдання по стенографуванню, отримані від адміністратора.

3. Редактор-коректор – виконує завдання, отриманні від адміністратора.

Для зовнішніх суб'єктів вищеприведені вимоги розділяються на три множини, які не перетинаються. Розділивши відповідним чином, множину функціональних аспектів системи на три частини, організуємо три підсистеми: АРМ адміністратора групи стенографістів (АРМ АГС), АРМ оператора-стенографіста (АРМ ОС), АРМ редактора-коректора (АРМ РК).

Підсистема АРМ АГС призначена для роботи з інформацією про мовний сигнал, що поступає, та керування процесом стенографування цього сигналу. З цією підсистемою взаємодіє зовнішній суб'єкт «Адміністратор групи стенографістів». Підсистема АРМ ОС призначена для роботи з фрагментами мовного сигналу з метою їх стенографування. Ця підсистема знаходиться під керуванням зовнішнього суб'єкта «Оператора-стенографіста». Підсистема АРМ РК призначена для редагування підсумкового документа, створеного операторами-стенографістами. Керувати двома підсистемами дозволено зовнішньому суб'єкту «Адміністратору групи стенографістів». Аспекти використання підсистем, виділені в результаті розподілу множини аспектів використання всієї ІС на три частини відповідають функціональності вищеприписаній та представлені на діаграмі використання (рис. 1).

Підсистеми АРМ АГС, АРМ ОС і АРМ РК зв'язані між собою: зміна даних в одній підсистемі може привести до зміни даних в іншій. Тобто між підсистемами має бути організований обмін повідомленнями. Для цього необхідно організувати сервер для реалізації функцій по обміну повідомленнями. Діаграма послідовностей (рис. 2) ілюструє схему обміну повідомленнями в системі.

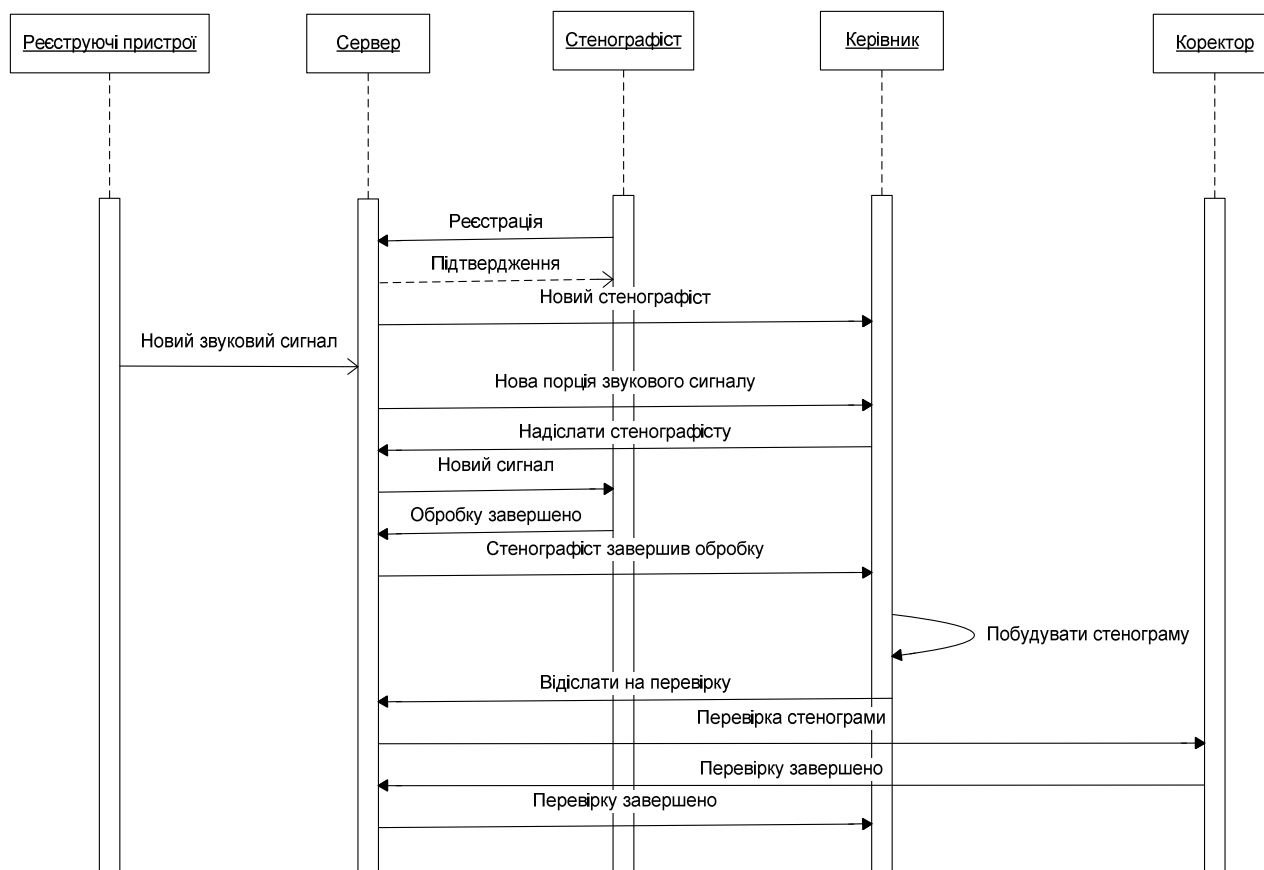


Рис. 2. Діаграма послідовностей для ІС

Логічна модель ІС

Структура кожного компонента ІС (кожної підсистеми та кожного аспекта використання) включає наступні елементи:

- єдина модель, яка описує організацію даних;
- види, які є візуальним представленням даних;
- інтерфейсні елементи, з допомогою яких змінюються дані, які зберігаються в моделі.
- Цим елементам відповідають наступні об'єкти в ІС:
- схема даних, яка складається із об'єктів-таблиць;
- запити, збережені процедури;
- керуючі запити, збережені процедури, форми для введення та зміни даних.

Модель даних. ІС має єдину модель даних (для всіх підсистем) яка призначена для збереження голосового сигналу та відповідної йому текстової інформації. Структура моделі даних впливає із розподіленого характеру обробки мовного сигналу та показана на рис. 3.

Модель представлення даних. Візуальним представленням даних, які зберігаються в моделі, є види, які змінюються тоді, коли змінюються ці дані. Для реалізації потрібної функціональності визначимо необхідні види:

- список фрагментів, яким призначений стенографіст;
- поточний фрагмент для вказаного стенографіста;
- список завершених стенограм;
- список, який містить інформацію про оброблені фрагменти та про відповідні їм стенограми;
- тексти фрагментів стенограми у правильному порядку;
- список неназначених фрагментів;
- ознака завершення стенограми.

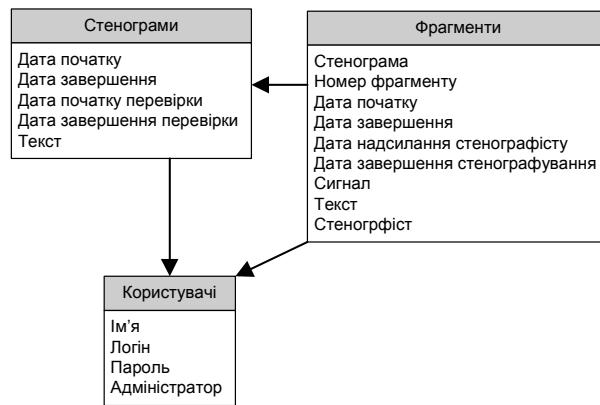


Рис. 3. Логічна модель даних для ІС

Модель керування даними. Побудова моделі керування даними базується на проектуванні інтерфейсу користувача та на бізнес-логіці організації даних у БД, реалізованій з допомогою збережених процедур. Елементи інтерфейсу користувача проектуються на основі діаграм активності для кожного аспекту використання та відображають необхідну функціональність ІС (рис. 4). Збережені процедури мають реалізувати наступну бізнес-логіку керування даними:

- додати рядок про нову стенограму;
- додати рядок про новий фрагмент стенограми;
- назначити стенографіста для фрагмента стенограми;
- завершити стенограму;
- завершити фрагмент стенограми.

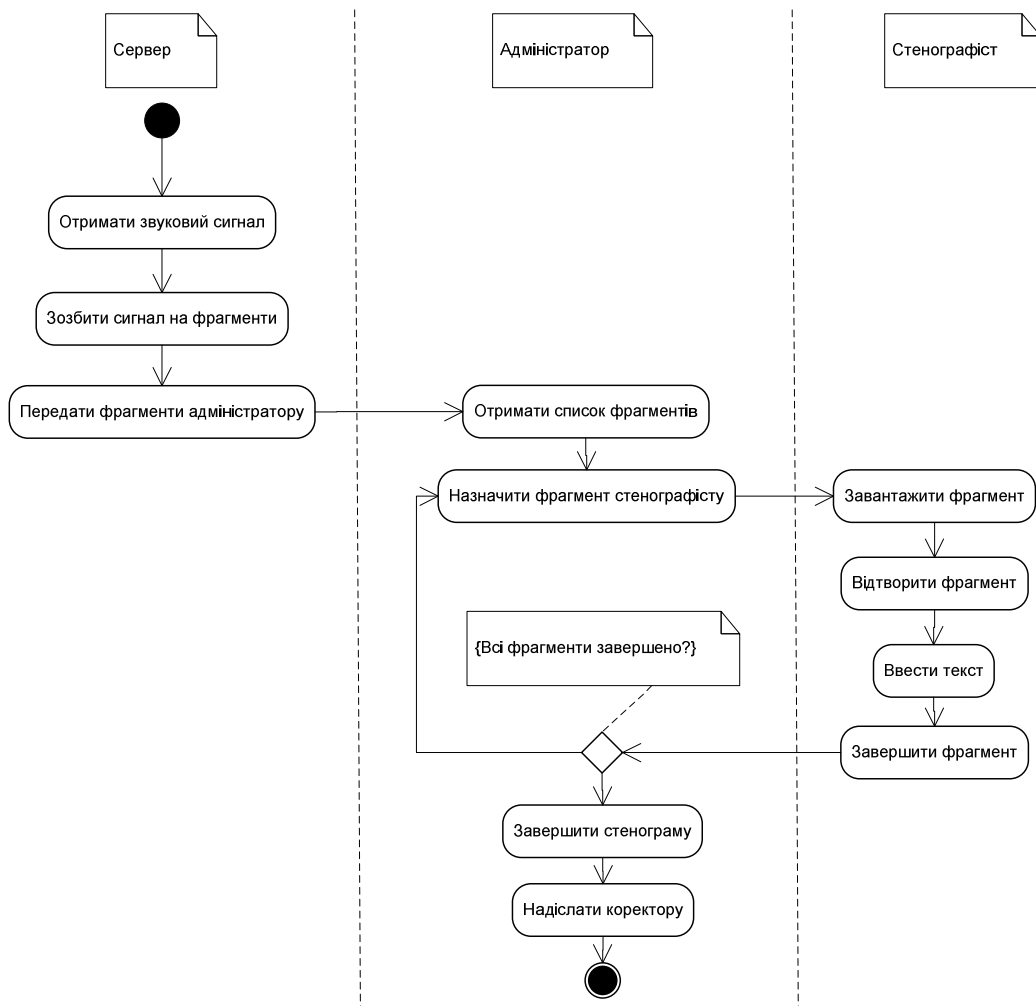


Рис. 4. Діаграма активності для отримання стенограми

Реалізація прототипу ІС

Прототип ІС реалізований з використанням технології DOT.NET, СУБД Microsoft SQL Server та триланкової архітектури (сервер БД, сервер ІС, три клієнти – АРМ АГС, АРМ ОС, АРМ РК).

Фізична модель даних включає в себе таблиці зі зв'язками (рис. 5) та збережені процедури (табл. 1):

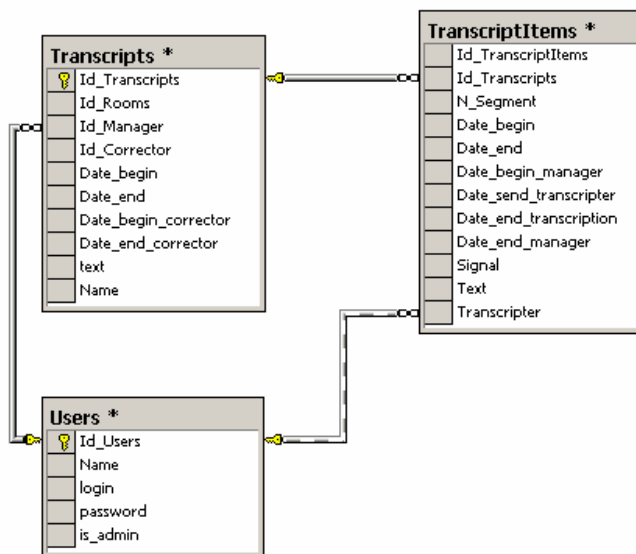


Рис. 5. Фізична модель даних ІС

Таблиця 1

Ім'я збереженої процедури	Опис
AddNewTranscript	Додає рядок про нову стенограму
AddNewTranscriptItem	Додає рядок про новий фрагмент стенограми
AssignTranscripiter	Призначає стенографіста для фрагмента стенограми
FinishTranscript	Завершує стенограму
FinishTranscriptItem	Завершує фрагмент стенограми
GetAssignedTranscriptItems	Повертає список фрагментів, яким призначений стенографіст
GetCurrentTaskByTranscripiter	Повертає поточний фрагмент для вказаного стенографіста
GetFinishedTranscripts	Повертає список завершених стенограм.
GetProcessedTranscriptItems	Повертає таблицю, яка містить інформацію про оброблення фрагментів та про відповідні їм стенограми
GetSegmentsText	Повертає тексти фрагментів вказаної стенограми у правильному порядку
GetUnassignedTranscriptItems	Повертає список неназначених фрагментів
IsTranscriptFinished	Повертає ознаку завершення стенограми
ReturnTranscriptItem	Повертає сигнал фрагмента стенограми
ReturnTranscriptItemText	Повертає текст фрагмента стенограми
SaveTranscriptItemText	Зберігає текст фрагмента стенограми

Для програмної реалізації ІС введена ієрархія основних класів, представлена на рис. 6.

Структурно ІС розділена на шість модулів: два службових приєднаних модулі Common та ServerCommon, в які винесені загальні класи для клієнтської та серверної частин ІС, та чотири виконуваних модулі, які реалізують сервер системи та три типи робочих місць: АРМ АГС, АРМ ОС, АРМ РК.

У модуль Common винесені три основних класи, які необхідні для роботи ІС. Клас AsynchronousClient реалізує роботу клієнтського сокета. Клас ClientSocketProcessor відповідає за обробку інформації, отриманої через клієнтський сокет. Клас SoundDB містить виклики збережених процедур у БД. Модуль Common містить абстрактний клас ClientForm, який слугує основою для реалізації форм клієнтських додатків системи.

Модуль ServerCommon містить основні класи, необхідні для роботи серверної частини ІС. Модуль ServerCommon також використовує класи, які зберігаються у модулі Common. Клас AsynchronousServer реалізує асинхронний серверний сокет. Клас ServerSoundDB містить виклики збережених процедур БД, необхідних для роботи серверної частини ІС. Клас SoundProcessor відповідає за обробку вхідного звукового сигналу та розбиття його на фрагменти. Модуль ServerCommon також містить абстрактний клас ServerForm, який слугує основою для реалізації форми серверного додатку системи.

Серверна частина системи реалізована на основі діалогового вікна, яке наслідує клас ServerForm. Аналогічно, АРМи реалізовані на основі форм, які наслідують клас ClientForm.

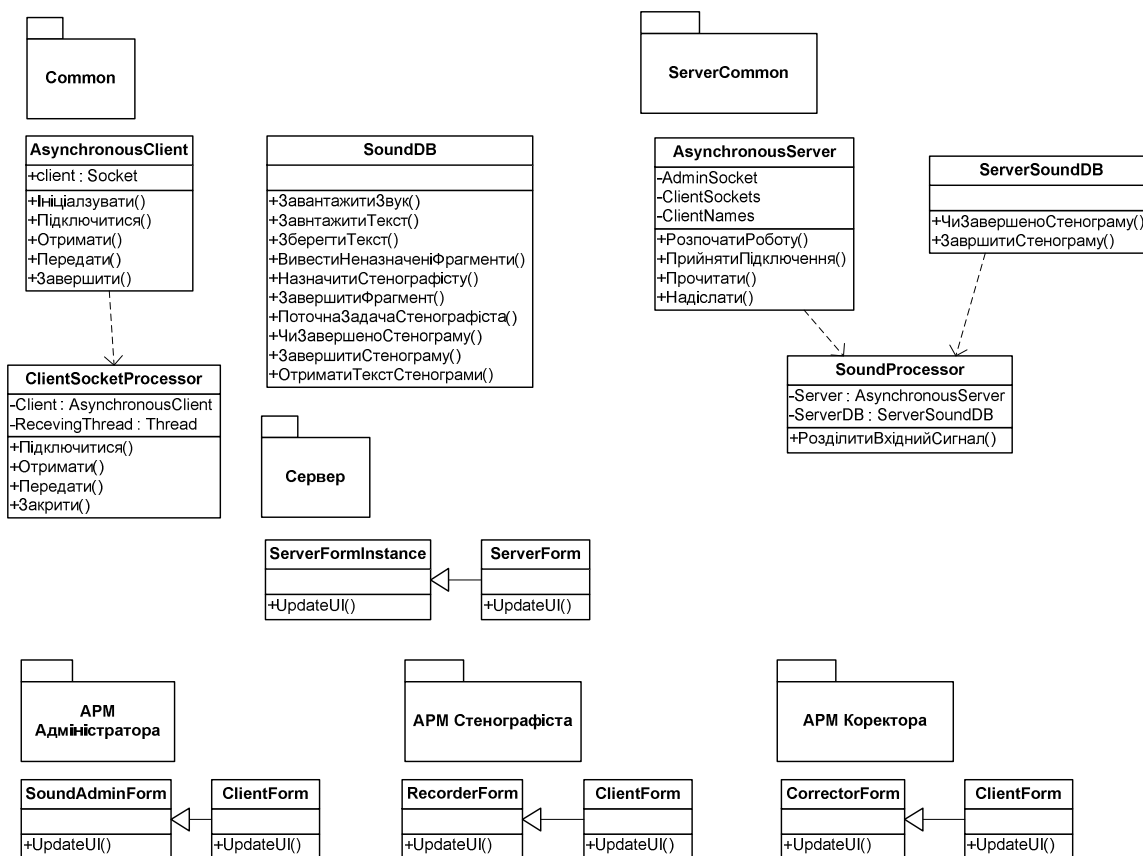


Рис. 6. Діаграма основних класів IC

Основні властивості та методи перерахованих класів приведені в табл. 2.

Таблиця 2

Властивості та методи	Опис
ClientSocketProcessor.Client	Об'єкт класу AsynchronousClient, який реалізує асинхронний клієнтський сокет
ClientSocketProcessor.ReceivingThread	Потік, у якому отримується і обробляється інформація від сервера
SoundDB.LoadSound()	Завантажити звуковий сигнал
SoundDB.LoadText()	Завантажити текст фрагмента
SoundDB.SaveText()	Зберегти текст фрагмента
SoundDB.ReturnUnassignedSegments()	Повертає список фрагментів, які не призначені стенографістам
SoundDB.AssignSegment()	Призначити фрагмент стенографісту
SoundDB.FinishSegment()	Звершити фрагмент
SoundDB.GetTaskByTranscriber()	Повертає ідентифікатор фрагмента, призначеного данному стенографісту
SoundDB.IsTranscriptFinished()	Повертає інформацію про завершення стенограми
SoundDB.FinishTranscript()	Звершує стенограму
SoundDB.GetTranscriptText()	Повертає текст стенограми
AsynchronousServer.ClientSockets	Масив клієнтських сокетів, які підключені до сервера
AsynchronousServer.AdminSocket	Сокет АРМ АГС
AsynchronousServer.ClientNames	Масив імен стенографістів, які підключені до сервера
SoundProcessor.SplitSignal()	Розбиває вхідний звуковий сигнал на фрагменти
ClientForm.UpdateUI() ServerForm.UpdateUI()	Віртуальні методи, які відповідають за асинхронне відображення інформації на формах серверної та клієнтських частин. Ці методи перевизначаються у класах ServerForm, SoundAdminForm, RecorderForm та CorrectorForm

Для демонстрації запропонованої технології реалізований прототип ІС (рис. 7), який включає у себе наступні компоненти ІС: СУБД, сервер ІС, АРМ АГС, АРМ ОС.

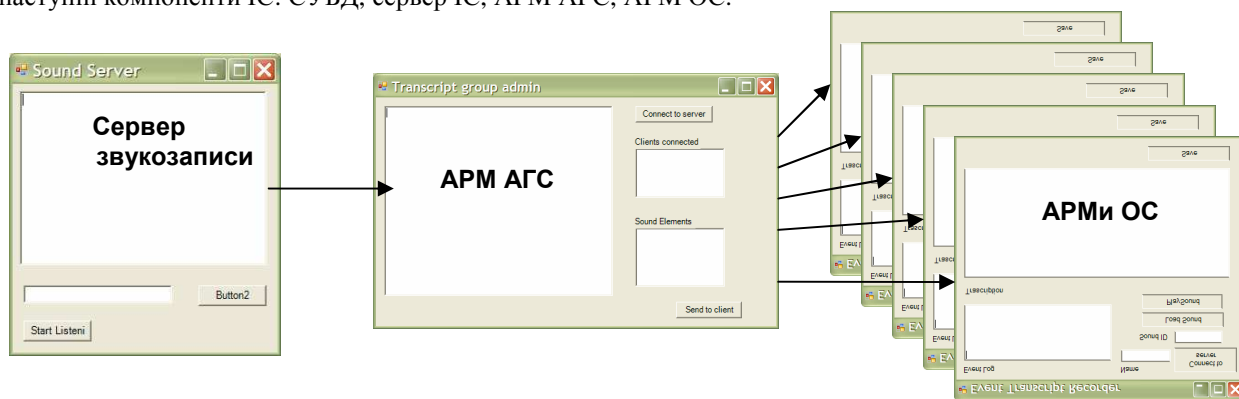


Рис. 7. Реалізація прототипу ІС

Висновки

Запропонована реалізація ІС розподіленого комп'ютерного документування є макетом системи, призначеної для автоматизації процесу стенографування мовних фонограм засідань. Для реалізації промислової версії ІС будуть проведені дослідження в наступних напрямках:

1. Загальні задачі для мовного сигналу:

- алгоритми сегментації неперервного мовного потоку на фрагменти, які містять мову та на паузи (з врахуванням послідовної обробки голосового потоку);
- алгоритми сегментації мовних фрагментів на пітчі (періоди основного тону) для української мови;
- алгоритми сегментації мовного фрагменту на ділянки, які не містять пітчів (шипячі тощо);
- алгоритми розпізнавання пітчів (для української мови);
- алгоритми розпізнавання непітчевих фрагментів;
- алгоритми розбиття мовних фрагментів (після розпізнавання пітчів та непітчів) на слова української мови;
- алгоритми уточнення (повного розпізнавання) слів з допомогою словника української мови та словника типових виразів української мови;
- алгоритми ідентифікації диктора для цілісних мовних фрагментів;
- алгоритми очищення від сторонніх шумів цілісних мовних фрагментів.

2. Специфічні задачі для проблеми автоматизації процесу стенографування

- оптимальне розбиття вхідного сигналу на рівноцінні фрагменти;
- автоматична побудова оптимальної черги фрагментів;
- оптимальний автоматичний розподіл фрагментів між вільними стенографістами;
- підтримка великої кількості залів, для яких ведеться стенографування;
- підтримка словників типових фраз та виразів для прискорення процесу вводу тексту стенографістом.

Отримані результати досліджень будуть імплементовані в фінальну версію ІС.

1. Система распределенного компьютерного документирования устных выступлений и фонограмм речи Нестор // <http://www.speechpro.ru/rus/products/doc-systems/nesstor/>
2. Комплекс оперативного стенографирования "SRS Report 2002" // <http://srs.kiev.ua/index.php?pg=2&lang=rus&tov=23>
3. The meeting recorder project // <http://www.icsi.berkeley.edu/Speech/mr/mtgrcdr.html>
4. Metzger F., Jin Q., Fugan C., Laskowski K., Pan Y., Schultz T. Issues in Meeting Transcription. – The ISL Meeting Transcription System // http://isl.lira.uka.de/fileadmin/publication-files/islMeetingSystem_icslp04.pdf
5. Yu H., Tomokiyo T., Wangand Z., Waibel A. New Developments in Automatic Meeting Transcription // Proceedings of ICSLP2000, 2000. <http://www.is.cs.cmu.edu/papers/speech/ICSLP2000/ICSLP2000-hua1.pdf>
6. Hain T., Burget L., Dines J., Garau G., Karafiat M., Linkoln M., Vepa J., Wan V. The AMI Meeting Transcription System: Progress and Performance, 2006. // <http://www.cstr.ed.ac.uk/downloads/publications/2006/AMIsr.nist06.pdf>
7. Yu H., Clark C., Malkin R., Waibel A. Experiments in automatic meeting transcription using JRTK. // Acoustics, Speech and Signal Processing, 1998. Proceedings of the 1998 IEEE International Conference. – 1998. 12-15 May. – Vol. 2. – P. 921 – 924.