System Documentation

David Nadrchal (12213656)
Giray Unlu (12338078)

June 23, 2025

Contents

1. Goal of the System

This project presents **Idea Decoder**, an automatic speech recognition (ASR) system designed to support individuals with severe speech impairments, often accompanied by motor disabilities such as quadriplegia that place additional constraints on how the system can be interacted with. The current prototype is tailored for a single real-world user — a Czech-speaking teenage girl who has lived with a permanent tracheostomy, quadriplegia and possibly dysarthria since early childhood, when she was struck by a car. Her speech is unintelligible to most listeners unfamiliar with her condition, posing significant communication barriers.

In this project, we put Whisper-based ASR developed by TracheoSpeech_ASR project into a real-world use. Our framework can be however used straightforwardly for other individuals.

Specifically, we set the following goals for our system:

1.1 Non-AI Goals

- Build a personalised ASR system for patients with various kinds of speech impediments, including but not limited to post-stroke conditions or permanent tracheostomy.
- Support patients in communicating effectively with unfamiliar individuals.
- Enable functional participation in everyday contexts such as school or the workplace.
- Allow voice-based control of smart devices to promote autonomy and improve quality of life.

1.2 AI-Related Goals

- Accurately recognize the patient's voice among other sounds.
- Learn and interpret the unique pronunciation patterns of individual patients and convert them into readable text.

2. Requirements

The system requirements are documented using the EARS (Easy Approach to Requirements Syntax) template, grouped into functional and non-functional categories. Additionally, we highlight the AI-related requirements that influence the core technical design.

2.1 Functional Requirements

- **Req1:** When the patient speaks, the Idea Decoder shall transcribe their words and display them on the screen.
- **Req2:** The Idea Decoder shall collect the patient's speech data and use it to improve recognition through further learning.
- **Req3:** While the system is open, the speaker recognition module shall process incoming audio and determine whether the patient is speaking.
- **Req4:** If the patient is detected as speaking, the speaker recognition module shall pass the audio input for transcription.
- **Req5:** When activated by the speaker recognition module, the ASR system shall transcribe the audio input.
- **Req6:** When the ASR system provides a transcription, the user interface shall display it on the screen.
- **Req7:** When a user unsubscribes, the Idea Decoder shall delete all associated data and any models trained on it.
- **Req8:** The Idea Decoder shall provide an interface to configure task-specific setups and vocabularies optimized for particular conditions or contexts.

2.2 Non-Functional Requirements

- **NfReq1 (External Interface):** The Idea Decoder shall provide a web-based user interface, with speaker recognition and ASR systems running on a server.
- **NfReq2 (Performance):** When called to transcribe an audio segment of length X seconds, the ASR system shall return the transcription in less than $4X$ seconds.
- **NfReq3 (Performance):** The speaker recognition system shall process an audio segment of length X seconds in less than X seconds.
- **NfReq4 (Accuracy):** The ASR system shall maintain a real-world word error rate lower than 20%.
- **NfReq5 (Accuracy):** The speaker recognition system shall achieve both precision and recall above 90%.
- **NfReq6 (Privacy and Ethics):** If and only if a person has a signed agreement with the patient, the Idea Decoder shall allow them to inspect transcribed segments and update the ASR model.

- **NfReq7 (Technical Constraint):** The ASR system shall be implemented in Python.
- **NfReq8 (Accessibility):** The Idea Decoder shall require only minimal tactile interaction for user control.

2.3 AI-Related Requirements

The following requirements are explicitly dependent on AI technologies:

- **Req1, Req2, Req5 – Speech-to-Text (Whisper by OpenAI):**
  By using the training interface, the system collects aligned audio-text pairs, which are then used to fine-tune the model. This allows accurate transcription of the patient's speech onto the screen.
- **Req1, Req3 – Speaker Activity Detection:**
  A separate model flags whether the patient is actively speaking. This is achieved by collecting representative background and patient-specific sound samples, enabling the system to differentiate the patient's voice from other audio sources.
- **Req3 – GPU-Powered Backend:**
  To ensure real-time speaker detection and transcription, the system backend leverages GPU acceleration, allowing it to efficiently handle streaming inputs and model inference.
- **Performance-Oriented Non-Functional Requirements:**
  NfReq2, NfReq3, NfReq4, NfReq5 concern the speed and accuracy of AI models used in the system.
- **Ethical and Privacy-Oriented Non-Functional Requirement:**
  NfReq6 addresses access control and model retraining policies aligned with patient consent.

3. Use Cases

We consider the following use cases:

Main Use Cases

- **A Dialogue:** Enables real-time conversation by detecting, transcribing, and displaying the patient's speech to all participants.
- **A Monologue:** Allows the patient to deliver extended speech, primarily one-sided, to a listener or voice-controlled system.
- **Ordering Groceries at a Local Shop:** Transcribes speech to assist in real-world interactions with shop clerks.
- **Having a Conversation at a Public Event:** Supports natural conversation in noisy or multi-speaker public environments.
- **Controlling a Home Smart Device by Voice:** Enables voice-based control of home appliances for increased autonomy.
- **Attending a Job or School Interview:** Facilitates real-time understanding in formal interview settings.
- **Unsubscribing and Data Deletion:** Allows the user to exit the system and have their data and models deleted.

Supporting Use Cases

- **Speaker Identification:** Detects whether the audio input belongs to the patient before triggering transcription.
- **Continuous Learning from User Speech:** Collects and uses new data to improve speech recognition accuracy over time.
- **Vocabulary Setup:** Allows configuration of task-specific vocabularies for better domain-specific performance.
- **Manage GPU Resources:** Allocates and schedules GPU usage to maintain performance and responsiveness.
- **Manage Authorized Access:** Controls permissions for who may view or edit transcriptions and model data.

Out of the identified use cases, **all but three are supported by our prototype**.

- **Manage Authorized Access** is not implemented at all. For our current patient, we prioritised ease of use, and incorporating authentication or account management would have added unnecessary complexity. Moreover, the security risks are currently minimal, as the application is primarily used for collecting training data (generated by repeated scripted dialogues of two characters) and does not process or store sensitive information.

- **Unsubscribing and Data Deletion** is not directly supported through the application interface, but can be easily performed manually upon request by the patient or their caretakers.
- **Controlling a Home Smart Device by Voice** is not supported, as this would require a text-to-speech component that is currently not part of the system.

3.1 Description

Here we provide an exhaustive list of our use cases with their success scenarios, linked use cases and other lists.

3.1.1 Main Use Cases

Use case: A Dialogue		
ID	A Dialogue	
Description	A patient with severe speech impediments wants to have a conversation with one or more other people, where they would speak in short sentences and speak about the same amount of time as their partners. The Idea Decoder detects their speech, transcribes it and displays the transcription both to the patient and to their conversation partners.	
Actors	Patient (with speech disabilities)	
Stakeholders:	Patient, Converstaion partners, regulators, course intructors	
Pre-Conditions	Running system, quiet environment	
Success end condition:	text appears on the device screen	
Failure end condition:	text doesn't appear / system gives failure	
Main Success Scenario		**Linked UCs**
1	1.The patient launches the Idea Decoder app.	
2	2.SUC1: Speaker Identification checks if it is indeed the patient speaking (rather than noise or a different speaker).	SUC1
3	3.The system transcribes what the patient says.	
4	4.The text appears on the device screen, large enough to be read by everybody involved in the conversation.	
5		
6		
Alternative Scenarios		
4.A1	1.The patient (or caretaker) pre-configures a vocabulary of their choice with certain items or brand names (see SUC3: Vocabulary Setup).	SUC3
4.A2	2.During the interaction, the system uses this specialized vocabulary, resulting in more accurate transcriptions for words of its topic.	
4.A3		
Exception Scenario		
3.A1	1.The system fails to confirm that the patient is speaking because of heavy background noise.	
3.A2	2.It displays an error or "No speech detected."	
3.A3	3.The patient repeats or tries again after moving to a quieter location.	

Use case: A Monologue		
ID	A Monologue	
Description	The patient wants to speak to a listener. The listener might be a voice-controlled machine or an actual person who sometimes says their own line, however most of the speaking is done by the patient.	
Actors	Patient (with speech disabilities)	
Stakeholders:	Patient, Listener, Patient's caretaker, regulators	
Pre-Conditions	Running system, quiet environment	
Success end condition:	text appears on the device screen	
Failure end condition:	text doesn't appear / system gives failure	
Main Success Scenario		**Linked UCs**
1	1.The patient starts speaking.	
2	2.The system confirms it's the patient's voice using SUC1.	SUC1
3	3.The system transcribes the patient's speech in real time.	
4	4.The text is displayed and possibly read aloud to the listener.	
5	5.The speech finishes successfully.	
6		
Alternative Scenarios		
4.A1	1.The patient (or caretaker) pre-configures a vocabulary of their choice with certain items or brand names (see SUC3: Vocabulary Setup).	SUC3
4.A2	2.During the interaction, the system uses this specialized vocabulary, resulting in more accurate transcriptions.	
4.A3		
Exception Scenario		
3.A1	1.The device battery runs out or the server is not reachable.	
3.A2	2.The system stops transcribing.	
3.A3	3.The caretaker or patient reboots or recharges the device if possible; if not, the speech must continue without the Idea Decoder.	

Use case: Ordering Groceries at a Local Shop		
ID	3	
Description	A patient with severe speech impediments wants to order groceries in-person at a local store. The Idea Decoder helps by detecting when the patient is speaking, transcribing the speech to text, and showing it on a device or a small screen that the store clerk can read.	
Actors	Patient (with speech disabilities)	
Stakeholders:	Patient, Store clerk, Regulators, System manager	
Pre-Conditions	Running system, quiet environment	
Success end condition:	text appears on the device screen	
Failure end condition:	text doesn't appear / system gives failure	

Main Success Scenario		**Linked UCs**
1	1.The patient walks up/is brought to the store clerk and launches the Idea Decoder app.	
2	SUC1: Speaker Identification checks if it is indeed the patient speaking (rather than noise or a different speaker).	SUC1
3	The system transcribes what the patient says (e.g., "I'd like two tomatoes, a loaf of bread, and some milk").	
4	The text appears on the device screen, large enough for the clerk to read.	
5	The clerk confirms the order.	
6		
Alternative Scenarios		
4.A1	The patient (or caretaker) pre-configures a "Grocery Vocab" with certain items or brand names (see SUC3: Vocabulary Setup).	SUC3
4.A2	During the interaction, the system uses this specialized vocabulary, resulting in more accurate transcriptions for specific grocery terms or brand names.	
4.A3		
Exception Scenario		
3.A1	The system fails to confirm that the patient is speaking because of heavy background noise.	
3.A2	It displays an error or "No speech detected."	
3.A3	The patient repeats or tries again after moving to a quieter location.	

Use case: Having a Conversation at a Public Event		
ID	4	
Description	A public social event (e.g., a conference or local festival). The patient wants to converse with multiple people around them, with the system automatically detecting when the patient speaks and transcribing it for those around.	
Actors	Patient (with speech disabilities)	
Stakeholders:	Patient, New acquaintances at the event , Regulators	
Pre-Conditions	Running system, quiet environment	
Success end condition:	text appears on the device screen	
Failure end condition:	text doesn't appear / system gives failure	

Main Success Scenario		**Linked UCs**
1	The event is crowded, but the system continually monitors incoming audio.	
2	SUC1: Speaker Identification identifies that the patient is speaking (versus other event chatter).	SUC1
3	The system quickly transcribes the patient's voice.	
4	Participants respond verbally, and the conversation flows.	
5	The clerk confirms the order.	
6		

Alternative Scenarios		
4.A1	1.The system requests the patient hold the microphone closer (minimizing background noise).	
4.A2	2.Transcription accuracy improves enough to continue the conversation.	
4.A3		

Exception Scenario		
3.A1	1.The caretaker tries to add a new "slang dictionary" mid-event.	
3.A2	2.The system cannot update instantly because GPU resources are unavailable (SUC4 or SUC5).	
3.A3	3.The caretaker is prompted to wait until the event ends or until the system can allocate resources.	

Use case: Attending a Job (or School) Interview		
ID	5	
Description	The patient attends a job or school interview where the interviewer needs to understand the patient's answers in real time.	
Actors	Patient (with speech disabilities)	
Stakeholders:	Patient, Interviewer , Patient's Caretaker, Regulators	
Pre-Conditions	Running system, quiet environment	
Success end condition:	text appears on the device screen	
Failure end condition:	text doesn't appear / system gives failure	
Main Success Scenario		**Linked UCs**
1	The interview begins, and the patient speaks.	
2	The system confirms it's the patient's voice using SUC1.	SUC1
3	The system transcribes the patient's speech in real time.	
4	The text is displayed for the interviewer, who can read and then proceed with the next question.	
5	The interview finishes successfully.	
6		
Alternative Scenarios		
4.A1	Before the interview starts, caretaker activates the custom dictionary in the Idea Decoder.	
4.A2	The system references that dictionary to improve recognition of job-specific terms.	
4.A3	The conversation flows with fewer recognition errors.	
Exception Scenario		
3.A1	1.The device battery runs out or the server is not reachable.	
3.A2	2.The system stops transcribing.	
3.A3	3.The caretaker or patient reboots or recharges the device if possible; if not, the interview must continue without the Idea Decoder.	

Use case: Unsubscribing and Data Deletion		
ID	7	
Description	The patient (or their legal representative) decides to stop using the Idea Decoder. They request that all personal data and model parameters be removed.	
Actors	Patient (with speech disabilities)	
Stakeholders:	Patient, System owners, Regulators	
Pre-Conditions	Running system, quiet environment	
Success end condition:	text appears on the device screen	
Failure end condition:	text doesn't appear / system gives failure	
Main Success Scenario		**Linked UCs**
1	1.The user opens the "Unsubscribe" option in the Idea Decoder settings.	
2	2.The system requests a confirmation.	
3	3.The user confirms, and the system permanently deletes the patient's recordings, transcripts, personal model derivatives, etc. (Req7).	
4	4.The system logs completion of unsubscription.	
5	5.The patient can no longer use the Idea Decoder.	
6		
Alternative Scenarios		
4.A1	1.The system packages transcripts, relevant model parameters, etc., into a downloadable file.	
4.A2	2.After export, the patient confirms final deletion.	
4.A3	3.The system removes the data from its servers.	
Exception Scenario		
3.A1	1.The system attempts to remove data from multiple locations but fails in one repository.	
3.A2	2.It notifies the caretaker/patient that partial deletion occurred.	
3.A3	3.The caretaker or user must contact support or retry until all data is removed.	

3.1.2 Supporting Use Cases

Supporting Use case: Speaker Identification		
ID	SUC1	
Description	Decides whether an audio should be passed for transcription	
Actors	Speaker Recognition System	
Stakeholders:	Patient, Conversation Partners, Listeners, Data Protection Regulators	
Pre-Conditions	Microphone is active	
Success end condition:	The patient's speech is recognized and passed to server, other sounds discarded	
Failure end condition:	The sound was not classified, transcription is not possible	
Main Success Scenario		
1	The system continuously listens for sound.	
2	When speech is detected, it compares the incoming voice to the patient's enrolled voice profile.	
3	If confidence > threshold, it flags the audio as "patient speaking" and triggers transcription.	
Alternative Scenario		
1.A1	The environment is too noisy.	
1.A2	The system recommends using a directional microphone approach to improve the match.	
Exception Scenario		
2.B1	The Speaker Recognition System fails or is unavailable (hardware issue).	
2.B2	The system logs an error and refuses to transcribe to avoid capturing the wrong person's speech.	

Supporting Use case: Continuous Learning from User Speech	
ID	SUC2
Description	Improves ASR accuracy for the patient by collecting data and scheduling training.
Actors	Patient, Caretaker, Tech admin, ASR system
Stakeholders:	Patient, Data Protection Regulators
Pre-Conditions	Patient consents to data collection and processing
Success end condition:	A new ASR model with lower WER is deployed
Failure end condition:	The training did not deliver an improvement. The old model remains to do ASR

Main Success Scenario	
1	The ASR system stores transcripts + audio pairs.
2	Periodically, caretaker or patient reviews & corrects transcripts.
3	The ASR system automatically schedules a fine-tuning (on a GPU if available)
4	New model is deployed for future sessions.
Alternative Scenario	
1.A1	The patient designates certain sessions as private
1.A2	ASR systém does not store the data for these sessions.
Exception Scenario	
3.B1	GPU or server resources for training are offline.
3.B2	The ASR system notifies tech admin, to run the training manually
3.B3	The ASR system sends an automated apology for the dealay to the patient/caretaker.

Supporting Use case: Vocabulary Setup	
ID	SUC3
Description	Patient/caretaker create specialized word lists for domain-specific recognition
Actors	Patient, caretaker
Stakeholders:	Patient, caretaker
Pre-Conditions	Patient plans a conversation on a very specific topic
Success end condition:	A vocabulary with purpose-relevant content is created
Failure end condition:	The provided vocabulary is invalid

Main Success Scenario	
1	Caretaker goes to “Manage Vocabulary” menu.
2	Caretaker enters or imports a list of domain-specific words, phrases, or names.
3	System stores the vocabulary.
4	When the conversation starts, the user activates the vocabulary.
5	ASR systém restricts the possible predictions to the words in dictionary.
Alternative Scenarios	
2.A1	The caretaker starts to typing in vocabulary forgetting that the vocabulary already exists.
2.A2	The System detects the duplicates.
2.A3	The System informs the caretaker that their work might be redundant.
Exception Scenario	
3.B1	System fails to load the new vocabulary file (invalid format)
3.B2	The caretaker must re-check or re-upload the file.

Supporting Use case: Manage GPU Resources	
ID	SUC4
Description	Ensure that computational resources are used efficiently
Actors	Tech admin, ASR system
Stakeholders:	Patient, Caretaker
Pre-Conditions	A training task is scheduled
Success end condition:	Training task is provided GPU space in terms of hours
Failure end condition:	The GPU is not available, the training cannot be executed

Main Success Scenario	
1	ASR system checks the GPU server is available.
2	The system runs training or advanced inference tasks on the GPU.
3	The tech admin regularly monitors resource usage, ensuring it completes without error.
Alternative Scenario	
1.A1	GPU is partially busy
1.A2	The system queues the job to start once resources free up.
1.A3	The system sends an automated apology for the dealay to the patient/caretaker.
Exception Scenario	
1.B1	The GPU is offline (driver or hardware issue).
1.B2	The ASR system notifies tech admin, to run the training manually
1.B3	The ASR system sends an automated apology for the dealay to the patient/caretaker.
1.B4	If tech admin does not start solving the problem fast, the system reverts to CPU-based training

Supporting Use case: Manage Authorized Access	
ID	SUC5
Description	Patient grants/revokes permissions to certain individual to view transcripts
Actors	Patient
Stakeholders:	Pateint, Caretaker, Patient's friends, teachers and other associates
Pre-Conditions	Patient decides to update the permissions
Success end condition:	Permissions were updated
Failure end condition:	The change of permissions was not authorized leading to errors

Main Success Scenario	
1	Patient or caretaker opens "Permissions" in the user interface
2	Patient authorizes or removes caretaker's ability to see transcripts or initiate training
3	The system updates the access control list
Alternative Scenario	
2.A1	The caretaker opens "Permissions" while granted only read-access
2.A2	The system denies the change of permissions
Exception Scenario	
3.B1	The caretaker attempts to read transcriptions after his permissions were revoked
3.B2	The system denies access and logs the attempt.

3.2 Use Case Diagram

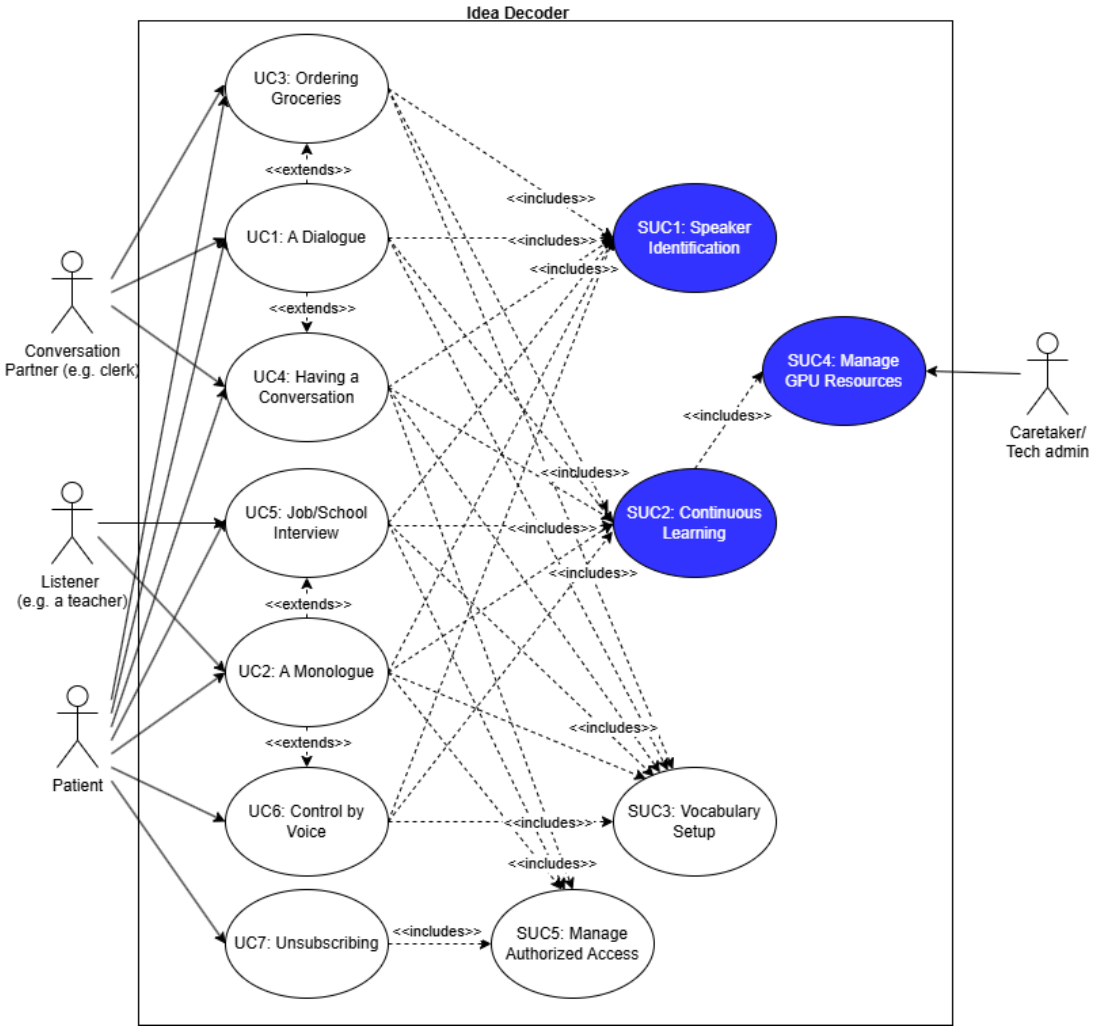

Figure 1: Use Case Diagram

3.3 Traceability Matrix

Use cases	UC1	SUC1	UC2	SUC2	UC3	SUC3	UC4	SUC4	UC5	SUC5	UC6	UC7
Requirements												
Req1	x		x		x		x		x		x	
Req2				x								
Req3	x	x	x		x		x		x		x	
Req4	x	x	x		x		x		x		x	
Req5	x		x		x		x		x		x	
Req6	x		x		x		x		x		x	
Req7												x
Req8	x		x		x	x	x		x		x	
NfReq1	x		x		x	x	x	x	x	x	x	x
NfReq2	x		x		x		x		x		x	
NfReq3	x	x	x		x		x		x		x	
NfReq4	x		x		x		x		x		x	
NfReq5	x	x	x		x		x		x		x	
NfReq6	x		x				x		x	x		x
NfReq7	x	x	x	x	x	x	x	x	x	x	x	x
NfReq8	x		x		x		x		x		x	x

Figure 2: Traceability Matrix

4. Domain Model

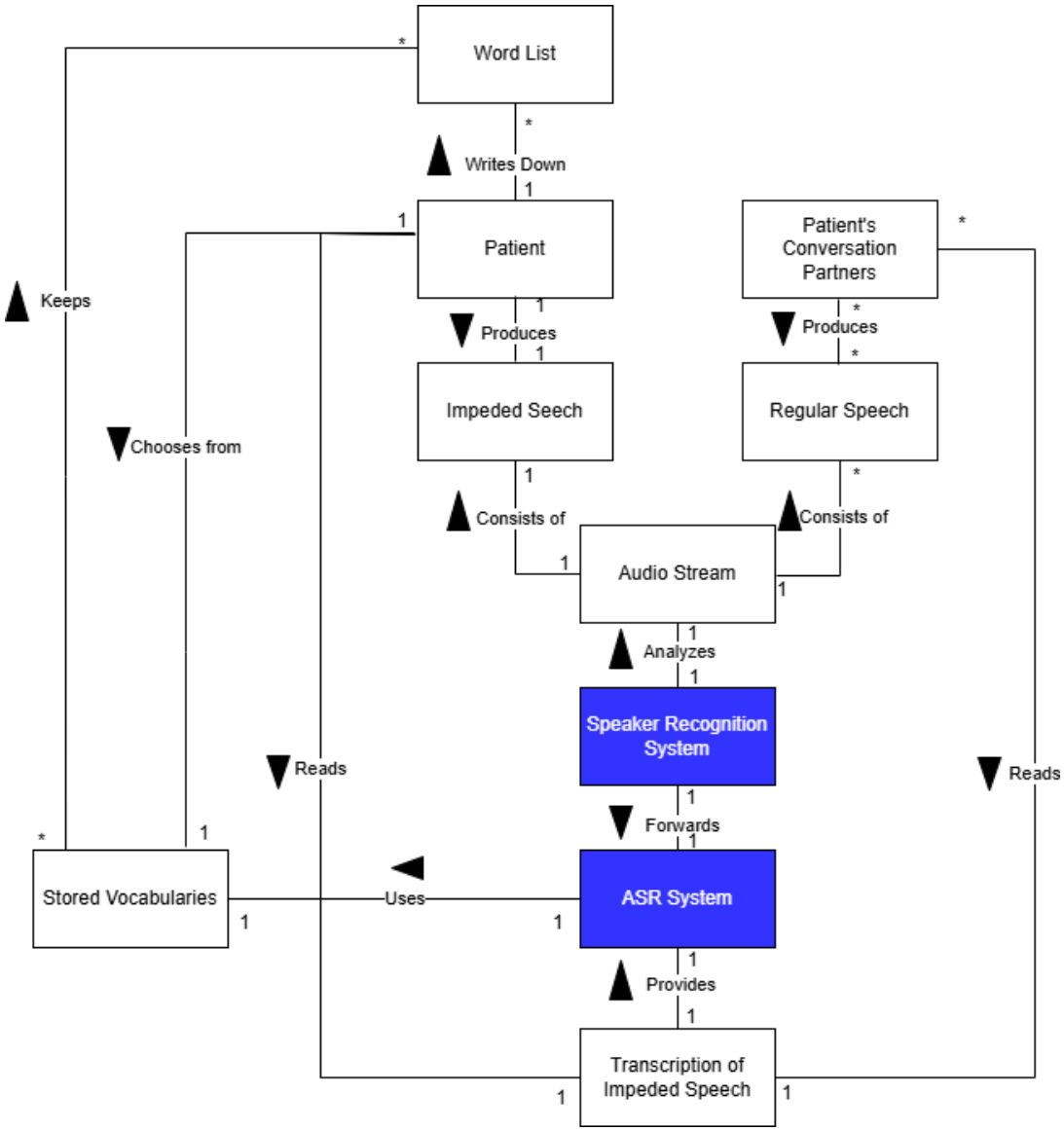

Figure 3: Domain Model

5. System Design

5.1 Architecture Diagram

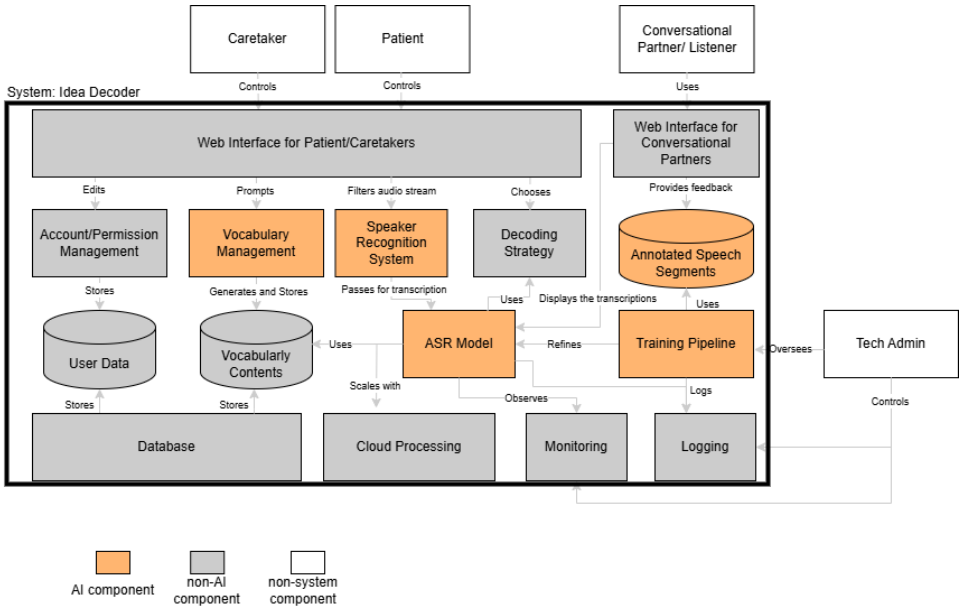

Figure 4: System Architecture

5.2 Component Descriptions

The following is a list of system components along with their functions and the requirements they fulfill. Except of *User Data* and *Account/Permission Management*, all of the components are implemented (however, for the Database part, we find it honest to note that our data storage is file-based and log-based and expanding the system for more users would probably require adopting more sophisticated and scalable solutions, namely SQL database).

- **Account/Permission Management:**
  Interface to manage access and edit rights for different users (e.g., adding or removing assistants). This component also handles unsubscribing.
  Related requirements: Req7, NfReq6

- **Annotated Speech Segments:**
  Collection of real-world audio segments paired with verified transcriptions via the web interface.
  Related requirements: Req2

- **ASR Model:**
  A Whisper-based model that receives audio segments and returns transcriptions. It is configured by a decoding strategy.
  Related requirements: Req1, Req4, Req5, NfReq1, NfReq2, NfReq4, NfReq7

- **Caretaker:**
  A user who can modify rights or vocabularies but does not use the system to

transcribe their own speech.
Related requirements: None

- **Cloud Processing:**
  Infrastructure used to run and train the ASR model efficiently.
  Related requirements: NfReq1, NfReq2

- **Conversation Partner/Listener:**
  An accountless user who engages in conversation with the patient. They view transcriptions and give feedback on their quality.
  Related requirements: None

- **Database:**
  Stores data that benefits from indexing, such as user data and transcription logs.
  Related requirements: Req7, NfReq6

- **Decoding Strategy:**
  Defines how the ASR model operates (e.g., beam size or logits masking).
  Related requirements: None

- **Logging:**
  Mechanism to record predictions and errors during both training and inference.
  Related requirements: None

- **Monitoring:**
  Tracks performance metrics (e.g., inference speed, memory usage) for review by the Tech Admin.
  Related requirements: NfReq2, NfReq3, NfReq4, NfReq5

- **Patient:**
  A user whose voice the ASR model is trained on. They also manage vocabularies and user accounts.
  Related requirements: None

- **Speaker Recognition System:**
  Processes audio input to detect when the patient is speaking and passes the relevant segments for transcription.
  Related requirements: Req1, Req3, Req4, Req5, NfReq1, NfReq3, NfReq5

- **Tech Admin:**
  A user who manually intervenes when system processes (especially training) require human supervision.
  Related requirements: None

- **Training Pipeline:**
  Fine-tunes the ASR model on the annotated segments. Requires high-performance computing; escalates to Tech Admin if unavailable.
  Related requirements: Req2, NfReq4, NfReq5

- **User Data:**
  Information about users of the system, especially access and edit privileges.
  Related requirements: Req7, NfReq6

- **Vocabulary Contents:**
  Domain-specific word lists used to constrain model predictions and improve accuracy.
  Related requirements: Req8

- **Vocabulary Management:**
  Tools to define standard conversational topics and corresponding vocabularies. May incorporate LLMs or word embeddings to assist.
  Related requirements: Req8

- **Web Interface for Conversational Partners:**
  Displays recent patient utterances and collects feedback on their accuracy and clarity.
  Related requirements: Req6, NfReq1, NfReq8

- **Web Interface for Patient/Caretaker:**
  Enables vocabulary, decoding strategy, and account management. Also allows activation of recognition. Requires login.
  Related requirements: Req6, Req8, NfReq8

5.3 Design Questions and Their Answers

Before implementing the system, we ask ourselves the following questions on potentially problematic aspects of our work and we propose solutions that could cope with them. Many of them are, however, not implemented at the moment.

- **How should the system behave if the patient's voice is extremely unclear or incomplete?**
  The system should detect low-confidence transcriptions using the ASR model's certainty scores and flag them for review. It may either request real-time feedback from the Conversational Partner or automatically suggest a clarification prompt.

- **What happens if a user forgets to update their vocabulary and keeps getting wrong predictions?**
  The system should monitor repeated low-confidence predictions and gently prompt the user (via the Web Interface for Patient/Caretaker) to update or expand their vocabulary to reflect recent conversation patterns.

- **How can the system adapt to rapid changes in a patient's speech style (e.g., due to illness progression)?**
  The Training Pipeline should allow fine-tuning on small, recent batches of Annotated Speech Segments to enable lightweight personalization without requiring full retraining.

- **What if a Conversational Partner gives wrong feedback intentionally or by mistake?**
  The system should monitor feedback patterns over time. If a partner consistently mislabels transcriptions, their feedback weight should be reduced during model retraining to preserve data integrity.

- **How should the system prioritize between model accuracy and real-time responsiveness?**
  The Decoding Strategy should support customizable profiles such as "fast but slightly less accurate" or "slow but very accurate," allowing the user to adjust trade-offs depending on session needs.

- **Can the system prevent a malicious user from overloading the cloud resources?**
  Yes. Account/Permission Management should enforce per-user rate limits, and Cloud Processing should implement throttling and queuing to prevent abuse.

- **How can the system ensure that a model personalized for one patient doesn't accidentally leak into another's session?**
  Each patient's ASR model should be sandboxed and isolated at the cloud level using secure session tokens and strict user identification to prevent crossover of vocabulary, weights, or settings.

- **How should feedback about system errors reach the Tech Admin quickly enough?**
  Critical issues detected via Logging and Monitoring should automatically trigger real-time notifications to the Tech Admin, including relevant diagnostic information.

- **How do we encourage caretakers to maintain the vocabulary over time?**
  Introduce friendly reminders and light gamification elements in the Web Interface for Patient/Caretaker to encourage vocabulary upkeep without adding cognitive burden.

- **Should the system display transcription uncertainties to the Conversational Partner, and if so, how?**
  Yes, but subtly. The UI should use soft visual cues—such as faded background highlights or small icons next to low-confidence words—to signal uncertainty without disrupting reading flow.