CS 491 - Senior Design Project I

Raporla.ai

Project Specification Document

Kerim Eren - 22103759

Ömer Amir Akdağ - 22102918

Ali Cevat Erçal - 22103341

Göktuğ Serdar Yıldırım - 22103111

Efe Tokar - 22103299

Table of Contents

1. Introduction

In the contemporary world, the health system has many challenges. Most of those challenges also apply to Turkey. When statements of the doctors from Turkey are evaluated, it appears that one of the biggest problems is that with the increasing number of patients, the time that doctors can spend per patient is severely limited. In order to fix this problem, automated technologies can be used.

One of the main solutions that many doctors need is the systems that can automatically make preliminary diagnosis. For example, for the fields like neurology or radiology, test results can be evaluated by image processing to make a preliminary diagnosis to help the doctors by reducing their time that is spent by evaluating the already evident findings. However, both the available data and necessary technology is not enough for this idea to become a senior project, even though it may be efficient.

So, our project Raporla.ai focuses on the other main problem that takes up doctors' time unnecessarily long: reports. Doctors must write formal report documents for their examinations and surgeries. These reports may take up unnecessary times since they should be written for every patient detailly. Especially in the fields like neurology, radiology, and pathology, the formal reports may be long and complex, therefore taking up a long time for doctors.

Furthermore, there is no standardized report format in any field. A doctor may write the same examination type’s report very differently from another doctor. This makes it difficult to understand and evaluate a patient's examinations by different doctors at different times.

By recognizing these challenges of the doctors from Turkey, Raporla.ai introduces an automated, speech-to-text report system that is specialized in Turkish and medical literature. By this, we aim to reduce the time spent on procedural things for doctors, and encourage them to focus on detailed examinations and treatments for the patients. Also, by introducing common templates, we aim to standardize the reports among different hospitals and different doctors. Last but not least, the application will also have a preliminary diagnosis suggestion system to create even more time for doctors by helping them to find out evident diagnosis in advance. This system will be purely based on medical datasets to minimize the model’s bias as much as possible.

1.1 Description

Raporla.ai aims to reduce the unnecessary time that doctors spend on procedural things. As mentioned in the introduction, especially in specific fields like neurology and radiology, writing reports may take a long time since doctors should examine the image results detailly. To this end, Raporla.ai introduces a speech-to-text model that is specialized in Turkish and specifically in Turkish medical literature. Beside these, our application will also have common report templates to bring standardized reporting systems between hospitals and doctors. In addition, there will be a suggestion system that suggests probable preliminary diagnosis to doctors.

By recognizing the importance of time management for doctors, the automated report generation system will dramatically reduce the time spent on reporting. Doctors will only record the examination report in audio. Afterwards, the model will detect the parts of the doctor's statements related to the fields of the report template. Finally, after making the inference, the model will place the necessary parts to the report template and will give the generated report in a formal format.

Furthermore, by introducing common report templates, the communication between the doctors by reports will be easier. There will be common report templates created for each field and each type of examination and surgery. Doctors also will be able to create their own templates and share with other doctors, if approved. The templates will be created, or approved, based on their coverage of the necessary report type. The report templates will be a union of all the necessary fields that must be stated for a particular examination or surgery. If there are empty fields that are not included in the doctor's statements, there will be warnings for them. If the doctor still finds them unnecessary, he/she may ignore the warnings and download the generated report.

In addition, Raporla.ai will have a preliminary diagnosis suggestion feature, too. After generating the report, the system will choose the keywords from the report's related fields. Then, it will search these keywords in a medical dataset, such as PubMed. Its information will be based on previous articles and reports from that medical dataset. By this, we aim to minimize the bias of the model's suggestions. After examining and evaluating the (a limited number of) relevant articles and reports, the system will give probable diagnosis to the doctor along with the sources it used for that particular diagnosis.

All these features aim to create more time for doctors and make them focus on special examinations and surgeries for each patient. Also, by introducing a standardized way of reporting, they aim to make communications between doctors more understandable and clear.

1.2 High Level System Architecture & Components of Proposed Solution

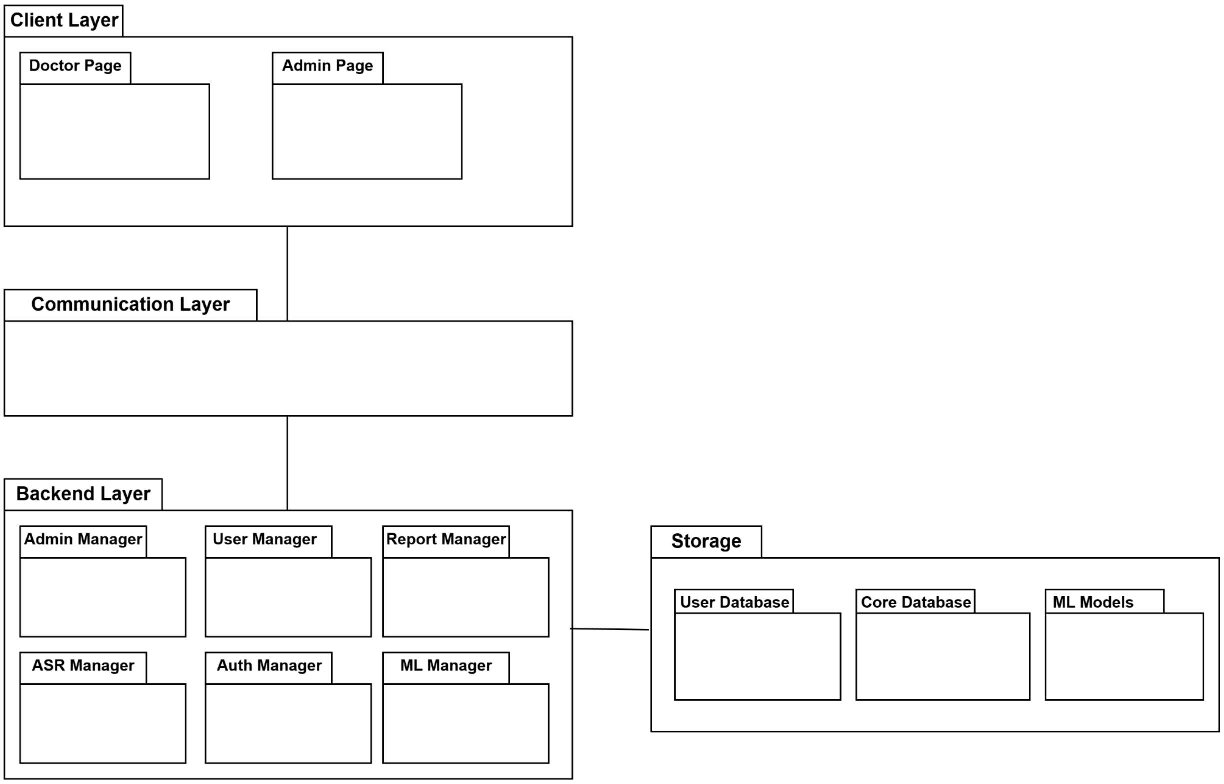

Figure 1: High level system architecture

1.2.1 Client Layer

The Client Layer is the user-facing component of the system where doctors and administrators interact with the platform through specialized interfaces.

- **Doctor Page:** The Doctor Page is designed for medical professionals to interact with the platform. It provides functionalities such as dictating reports via speech-to-text, reviewing generated reports, and managing patient-specific information. It ensures ease of use and seamless integration with clinical workflows.
- **Admin Page:** The Admin Page is designed for system administrators to monitor and manage platform operations. It allows administrators to see user activity, manage roles and permissions, update the system configuration, and provide access to logs and analytics.

1.2.2 Communication Layer

The Communication Layer serves as the intermediary between the client and backend layers. It handles requests from the client interface and ensures data exchange between components. Its purpose is to secure data transmission for speech-to-text processing, report generation, and user management. Another use case is to handle API integrations to connect client actions with backend services like authentication, report generation, and machine learning-based tasks.

1.2.3 Backend Layer

The Backend Layer is the core processing hub responsible for handling business logic and executing user requests. It uses several managers to modularize different functionalities.

- **Admin Manager:** Manages administrative operations, including overseeing system-wide settings, updating configurations, and managing access to backend functionalities. It serves as a critical tool for maintaining system integrity and scalability.
- **User Manager:** Handles all user-related functionalities, such as profile management, role assignments, and updating personal preferences. It also tracks user activity and ensures personalized system behavior.
- **Report Manager:** Responsible for generating and managing medical reports. It processes speech-to-text outputs, integrates patient data, and creates accurate, formatted reports for doctors to review.
- **ASR (Automatic Speech Recognition) Manager:** Processes audio input from doctors to convert speech into text. It uses fine-tuned machine learning models optimized for medical terminology to ensure high accuracy in transcriptions.
- **Auth Manager:** Manages user authentication and authorization processes. It handles tasks like login, password recovery, and session management, ensuring secure and reliable access to the platform.
- **ML Manager:** Manages machine learning models used across the platform. This includes model deployment, monitoring, and updating. It ensures the models remain efficient and relevant for speech recognition.

1.2.4 Storage Layer

The Storage Layer is the data management backbone of the system. It organizes and maintains all essential data required for system functionality.

- **User Database:** Stores user-specific information, including account details, preferences, and activity logs. It ensures quick and secure access for user-related functionalities.
- **Core Database:** The Core Database serves as the central repository for system-critical data, ensuring efficient organization and reliability. It stores all generated reports for doctors, making them easily retrievable for future reference or edits. Additionally, it holds system configuration settings and operational logs, which are vital for maintaining

and optimizing the platform's functionality. Admin-specific data, such as monitoring details and tools for updating system processes, are also housed here.

- **ML Models:** Contains trained machine learning models for Automatic Speech Recognition and other AI functionalities. It facilitates version control and efficient deployment of models across the system.

1.3 Constraints

1.3.1 Implementation Constraints

Technology Stack:

- **Frontend:** React.js [1] will be used for user interactions and interface design.
- **Backend:** Python FastAPI [2] framework will be used for server-side processes and API management.
- **Database:** MySQL [3] will be used for storing and managing system data. The database will be configured to meet performance and scalability needs.
- **Cloud Hosting:** The database and backend service will be hosted on AWS [4], enhancing reliability and scalability.
- **Machine Learning:** Model training and experimentation will utilize remote GPU resources on Google Colab [5].
- **Dataset Requirements:** The dataset will be manually created through a protocol with Eskişehir Osmangazi University and will include ethically approved data.
- **Testing Environment:** The system will be tested on Chrome browser to ensure compatibility with widely used platforms.
- **Integration:** Seamless integration with third-party tools such as PubMed will be ensured [6].

1.3.2 Economic Constraints

Cloud Services Cost:

- MySQL and backend services using FastAPI will be hosted on AWS or Google Cloud. The costs of these services will be optimized within the project budget limits.
- Cloud-based resources will be configured cost-effectively in terms of processing power and storage capacity.

Google Colab Usage:

- GPU resources from Google Colab's free version will be used to support model training. If necessary, a transition to Colab Pro will be made.

- Alternative, low-cost GPU solutions will be considered if prolonged GPU usage is required during training.

Scalability Costs:

- If the number of users increases, the costs of AWS or Google Cloud resources will also increase. Scaling strategies will be implemented, and the budget will be kept under control.
- Additional storage and processing power requirements will be addressed as the system expands.

Database and Storage Costs:

- As the MySQL database grows, cloud hosting costs will be optimized for increasing storage needs.
- Storage optimization and compression methods will be used to reduce costs.

1.3.3 Ethical Constraints

Data Security:

- Patient data will be stored encrypted and protected during transmission.
- The system will comply with data protection regulations such as KVKK [7] and GDPR [8].

AI Bias and Transparency:

- Dataset biases will be controlled, and a balanced dataset will be created to ensure the model works accurately and fairly for various scenarios and user groups.
- Diagnostic suggestions will only be provided as supportive guidance and not as definitive medical conclusions. This will be clearly communicated to users.

1.3.4 Temporal Constraints

- **Project Timeline:** The system will be developed and tested within the academic semester.
- **Model Training:** Machine learning models will be trained and optimized within the same academic period.
- **User Testing:** Time will be allocated for usability testing with healthcare professionals, and the system will be refined based on feedback.

1.3.5 Operational Constraints

- **Internet Dependency:** The system will require a stable internet connection for many features, which may create challenges in regions with limited access.
- **User Training:** The application will be developed to be simple enough for healthcare professionals to use with minimal training.

1.3.6 Legal and Regulatory Constraints

Medical Data Processing:

- The system will comply with KVKK for Turkey and GDPR for international regulations.
- Patient data will be anonymized using specific cryptographic methods.

1.3.7 Scalability Constraints

- **System Scalability:** The platform will be designed to support increased user loads and the addition of new medical fields.
- **Data Storage:** The MySQL-based data storage system will be designed to efficiently manage increasing data volumes.
- **Feature Expansion:** The architecture will allow for the addition of new features such as integration with hospital management systems or multilingual support.

1.3.8 Technical Constraints

- **Hardware Limitations:** Speech-to-Text and diagnostic suggestion models will be optimized to work on available server hardware.
- **Real-Time Processing:** The system will perform Speech-to-Text and report generation in real-time or near real-time.
- **Device Compatibility:** The application will be compatible with both web and mobile platforms.
- **Model Size:** Machine learning models will be lightweight to match the memory capacities of cloud servers.

1.4 Professional and Ethical Issues

Patient Privacy:

- The system will ensure the security of patient data through strong encryption methods.
- All data will be anonymized and accessible only to authorized personnel.
- Full compliance with KVKK and GDPR data protection regulations will be ensured.

Data Usage and Transparency:

- Medical reports and suggestions will only be provided as guidance and will not be regarded as definitive diagnoses. This will be clearly stated to users.

AI Bias:

- Potential biases in training datasets will be analyzed and corrected to achieve fair and impartial results.
- The model will be designed to avoid discrimination based on gender, age groups, or clinical conditions.

Impact of Errors:

- The system will emphasize that the reports and diagnostic suggestions it produces must be verified by doctors.
- Any misinformation or incomplete diagnosis suggestions will be carefully examined, and accountability will be clearly defined.

Ethical AI Usage:

- The system will include appropriate warning mechanisms to prevent users from over-relying on AI-generated reports and suggestions.
- Users will understand that the system is an assistive tool and that medical decisions should always be made by a qualified doctor.

Accessibility:

- The system will be designed to be accessible and easy to use for users with varying levels of technological expertise.
- The application will aim to serve without discrimination for any user group.

Data Sharing and Third Parties:

- Collected data will not be shared with third parties under any circumstances, except with user consent for a specific purpose.
- User consent will be obtained for all stages of data sharing.

Compliance with Standards:

- The system will be developed in compliance with international standards for medical reporting and data management.
- Ethical AI design guidelines such as those from the IEEE Global Initiative on Ethics of Autonomous and Intelligent Systems will be followed.

1.5 Standards

- **Data Security and Privacy:**
  - **KVKK and GDPR Compliance:** Personal and medical data of users will be processed in full compliance with KVKK in Turkey and GDPR in Europe.
  - **Encryption:** Data will be encrypted during transmission and storage.
- **Health Data Integration:**
  - **HL7 (Health Level Seven):** The system will comply with HL7 standards [9], internationally recognized for health data sharing.
- **Machine Learning Accuracy:**
  - **WER (Word Error Rate):** The Speech-to-Text model will be optimized using this metric to enhance accuracy [10].
- **Accessibility:**
  - **WCAG 2.1:** The system will be designed to meet these standards, ensuring easy access for users with disabilities [11].

2. Design Requirements

2.1 Functional Requirements

- The user can register the system with their credentials and login afterwards.
- The user can record their reports such as examination and surgery reports as audio files.
- The speech-to-text model will convert these recorded audio files to text files in Turkish and medical literature.
- The user can generate standardized reports using predefined templates based on the transcribed text.
- The system will warn the user if there are empty fields, before submitting the report.
- The user can create customized reports by removing unnecessary fields.
- The system provides anticipated diagnosis based on keywords taken from the report and verified against medical datasets like PubMed.
- The user can download the report.
- The user can delete an existing report.
- The user can delete their account.
- The user can search an existing report
- The user can select a report type that already exists.

2.2 Non-Functional Requirements

2.2.1 Usability

- The interface should be easy for common usage and require minimal training for doctors to use effectively.
- The application should be compatible with Turkish language and medical terminology [12].

2.2.2 Reliability

- The system should be performed under any condition without crashing or failing.
- Speech-to-text conversion should maintain high accuracy even with background noise or variation in voices [12].

2.2.3 Performance

- The application should process audio files and generate reports to create an ideal experience.
- Diagnosis suggestions should be retrieved and displayed within feasible and reasonable seconds after report generation [12].

2.2.4 Supportability

- The application should be easy to update for incorporating new medical datasets and templates [12].

2.2.5 Scalability

- The application must handle increasing numbers of users and simultaneous audio uploads without significant performance decline.
- The application should support the addition of new features like integration with hospital management systems [12].

3. Feasibility Discussions

3.1 Market & Competitive Analysis

In the market, several AI-powered solutions address problems similar to those that Raporla.ai aims to solve. However, none fully align with our target audience's needs or our feature set. One of the most notable examples is RadMate AI [13], which assists radiologists in generating reports through speech-to-text technology. While this platform shows the practicality of automating medical documentation, it is limited to radiology and designed only for English-speaking users.

Hence, it falls short of meeting the demands of Turkish-speaking medical professionals. Moreover, RadMate AI does not offer the ability to reference medical literature for preliminary diagnostic suggestions, a planned feature of Raporla.ai.

Another relevant tool is Amazon Transcribe Medical [14], which offers high-accuracy transcription for healthcare applications. However, this service is not optimized for Turkish language or practices, making it less suitable for the needs of Turkish doctors. Additionally, it lacks the capability to generate standardized reports or provide diagnostic support using medical databases, leaving room for Raporla.ai to fill this gap.

Similarly, Rad AI [15] offers advanced solutions for automating radiology impressions and follow-ups, but its lack of support for languages other than English and its focus on specific radiology tasks make it less adaptable to a broader Turkish medical context. In conclusion, while existing solutions prove the feasibility of integrating AI into medical reporting, they do not cater to the specific needs of Turkish doctors. Raporla.ai's unique focus on Turkish medical literature, standardized reporting, and as a future feature, preliminary diagnosis, makes it a promising and competitive project in this space.

In summary, while these existing tools illustrate the feasibility of integrating AI into healthcare documentation, they do not fully meet the specific needs of Turkish medical professionals. Raporla.ai differentiates itself by specializing in Turkish medical terminology, offering standardized templates, and providing diagnosis suggestions, making it a competitive and innovative solution.

3.2 Academic Analysis

Research in the fields of speech-to-text and medical AI provides strong academic backing for Raporla.ai's core functionalities. Studies on automatic speech recognition (ASR) demonstrate the feasibility of achieving high transcription accuracy using fine-tuned machine learning models trained on specialized datasets. For instance, research involving ASR models optimized for non-English languages, including Turkish, highlights the growing availability of tools capable of handling the nuances of localized languages and technical, in our case medical, jargon.

In terms of preliminary diagnosis, the integration of natural language processing (NLP) and machine learning with medical literature has been shown to assist healthcare professionals in identifying probable conditions. Systems that extract relevant keywords and reference structured databases, such as PubMed [6], have demonstrated significant accuracy in providing contextually appropriate suggestions. These findings validate the academic basis for implementing such a feature in Raporla.ai.

Finally, academic literature frequently emphasizes the value of reducing repetitive tasks for healthcare professionals. Automating processes such as report generation not only alleviates physician workload but also allows more time for patient care, thereby improving outcomes.

These findings show that Raporla.ai draws upon well-established academic research, validating its functionality and relevance. By addressing documented gaps in existing systems, the project is positioned to make a meaningful contribution to medical technology.

4. Glossary

ASR (Automatic Speech Recognition): A technology that converts spoken language into text using machine learning algorithms, specifically optimized in this project for Turkish medical terminology.

Core Database: The central repository where essential data, such as user profiles, reports, and system configurations, are securely stored and managed.

Data Encryption: The process of converting data into a coded format to prevent unauthorized access, ensuring secure transmission and storage of sensitive information like patient data.

Keyword Extraction: The process of identifying key terms from generated reports for use in diagnosis suggestions or further analysis.

ML (Machine Learning): A subset of artificial intelligence focused on training algorithms to make predictions or decisions without being explicitly programmed.

Real-Time Processing: The system's capability to convert speech into text and generate reports almost instantly, enhancing efficiency in medical workflows.

Speech-to-Text: A feature of the system that converts doctors' spoken words into written text, designed to streamline report generation.

5. References

[1] "React – A JavaScript library for building user interfaces," reactjs.org. https://reactjs.org. (accessed Nov. 21, 2024).

[2] "FastAPI," fastapi.tiangolo.com. https://fastapi.tiangolo.com. (accessed Nov. 21, 2024).

[3] "What is MySQL?," MySQL 8.0 Reference Manual. https://dev.mysql.com/doc/refman/8.0/en/what-is-mysql.html. (accessed Nov. 21, 2024).

[4] "AWS Free Tier," aws.amazon.com. https://aws.amazon.com/tr/free/. (accessed Nov. 21, 2024).

[5] "Google Colab," colab.research.google.com. https://colab.research.google.com/. (accessed Nov. 21, 2024).

[6] "PubMed," pubmed.ncbi.nlm.nih.gov. https://pubmed.ncbi.nlm.nih.gov/. (accessed Nov. 21, 2024).

[7] "Kişisel Verileri Koruma Kurumu Resmi Web Sitesi," kvkk.gov.tr. https://www.kvkk.gov.tr/. (accessed Nov. 21, 2024).

[8] "General Data Protection Regulation (GDPR)," gdpr-info.eu. https://gdpr-info.eu/. (accessed Nov. 21, 2024).

[9] "HL7 Standards - Health Level Seven International," hl7.org. https://www.hl7.org/. (accessed Nov. 21, 2024).

[10] "What is Word Error Rate (WER)?," clari.com. https://www.clari.com/blog/word-error-rate/#:~:text=Put%20simply%2C%20WER%20is%20the,are%20a%20little%20more%20nuanced. (accessed Nov. 21, 2024).

[11] "Web Content Accessibility Guidelines (WCAG) 2.1," w3.org. https://www.w3.org/TR/WCAG21/. (accessed: Nov. 21, 2024).

[12] A. Papworth, "Non Functional Requirements: Quick Guide for the business analyst in 2024," BusinessAnalystMentor.com. https://businessanalystmentor.com/non-functional-requirements/ (accessed Nov. 21, 2024).

[13] "RadMate AI," radmate.ai. https://www.radmate.ai/ (accessed Nov. 20, 2024).

[14] "Amazon Transcribe Medical," aws.amazon.com. https://aws.amazon.com/transcribe/medical/ (accessed Nov. 20, 2024).

[15] "Rad AI," radai.com. https://www.radai.com/ (accessed Nov. 20, 2024).