

MedixClarity: A Privacy-First Platform for Automated Medical Report Analysis Using Local AI

Vasile-Răzvan Marin

Faculty of Computer Science,
Alexandru Ioan Cuza University
of Iași
Iași, Romania
razvan.vasile14@yahoo.com

Adrian Iftene

Faculty of Computer Science,
Alexandru Ioan Cuza University
of Iași
Iași, Romania
adiftene@gmail.com

Dragoș Silion

Faculty of Computer Science,
Alexandru Ioan Cuza University
of Iași
Iași, Romania
silidragos@gmail.com

ABSTRACT

Medical laboratory reports contain numerous parameters, numerical values, measurement units, and reference intervals whose interpretation may be difficult for users without specialized medical knowledge. This paper presents MedixClarity, a web platform for automated medical report analysis using self-hosted artificial intelligence. The processing pipeline combines PaddleOCR PP-OCRv4 for text extraction, LLaMA 3.1:8b for structured medical data parsing and interpretation, a six-step post-extraction validation mechanism, and a lexical retrieval-augmented generation system containing over 230 medical terms. Medical report content is not transmitted to third-party AI inference APIs; instead, AI services are deployed on cloud compute instances controlled by the application. Evaluation on two real medical reports illustrates the behavior and limitations of the pipeline, while an exploratory usability study with 12 participants yielded an overall mean score of 4.87/5. The results demonstrate the technical feasibility of the approach while identifying important limitations concerning demographic context, retrieval quality, and AI-generated interpretations.

AUTHOR KEYWORDS

medical report analysis; large language models; retrieval-augmented generation; OCR; privacy-oriented AI; LLaMA; PaddleOCR

DOI: 10.37789/icusi.2026.31

INTRODUCTION

The digitalization of healthcare systems has generated an increasing volume of medical documents and reports in electronic or scanned formats, whose correct interpretation remains a significant challenge for patients without specialized medical training. A laboratory report may contain dozens of parameters with numerical values, measurement units, and reference intervals, while understanding their significance often requires consulting a physician or individually researching unfamiliar medical terminology. Although medical interpretation should ultimately remain under the supervision of qualified healthcare professionals, patients may benefit from accessible tools that help them better understand the structure and meaning of their laboratory results before or after a medical consultation. Such tools can reduce the informational barrier between patients and medical data and support more informed discussions with healthcare professionals.

The widespread adoption of large language models (LLMs) has opened new perspectives for processing and interpreting medical text [16, 27]. However, many existing solutions

rely on external cloud AI services, raising concerns regarding the processing of sensitive medical data protected under the European General Data Protection Regulation (GDPR) [7]. Transmitting medical reports to external AI services may expose sensitive health information to additional processing environments and infrastructure providers, creating privacy and data-governance challenges that are not always transparent to end users.

MedixClarity addresses both challenges: the difficulty patients face in understanding medical reports and the need to minimize unnecessary exposure of sensitive medical data. The platform provides users with an accessible and transparent way to explore the content of their laboratory reports while keeping AI inference under the control of the application infrastructure. The use of a self-hosted language model, combined with a microservices architecture [13] and a privacy-oriented processing pipeline, represents the technical response to the data-protection challenge, while the user interface, multilingual support, and interactive explanations address the accessibility needs of end users.

The main contributions of this work are: (1) an OCR service based on PaddleOCR PP-OCRv4 with conditional image pre-processing; (2) a Parser service using LLaMA 3.1:8b with prompt engineering and a six-step anti-hallucination validation mechanism; (3) a lexical RAG system with over 230 medical terms and more than 1500 aliases, achieving 10.93 ms average retrieval latency in the evaluated setup; (4) a REST API backend with JWT authentication; and (5) a React frontend with multilingual support and interactive medical-report exploration.

The remainder of this paper is organized as follows. The Related Work section discusses existing applications and relevant research, while the Proposed Solution section presents the architecture and its main components, including OCR, structured parsing, RAG, and privacy mechanisms. The Experiments and Evaluation section describes the technical evaluation, usability study, performance measurements, and identified limitations, followed by conclusions and future development directions.

RELATED WORK

Prior to designing MedixClarity, an analysis of existing solutions in the digital healthcare domain was conducted, focusing on applications that use artificial intelligence for symptom interpretation or medical document analysis. The initial selection placed particular emphasis on Romanian solutions because MedixClarity was designed primarily for Romanian-speaking users and for medical reports produced by Romanian healthcare providers. This local focus was complemented by

international reference platforms in order to position the proposed system within a broader digital-health context.

Six existing applications were selected for comparison: two developed by Romanian healthcare providers (Synevo Decoder and Regina Maria Virtual Assistant), two created by Romanian-speaking development teams (SymptoChat and Cindi), and two international reference platforms (Ada Health and Medicus AI Smart Reports). Table 1 compares these six existing applications with MedixClarity, which is included as the proposed system rather than as an additional related-work application.

Synevo Decoder is a virtual consultation service for interpreting medical analysis results, launched in December 2021 by Synevo Romania in partnership with the ATLAS platform [25]. The service provides virtual consultations with specialist physicians and is available to patients who have performed tests within the Synevo network. Compared with MedixClarity, Synevo Decoder relies on human physicians for interpretation and operates within the Synevo healthcare ecosystem.

SymptoChat is a Romanian AI-based medical pre-diagnosis mobile application [4]. It processes user-described symptoms through generative AI and provides possible conditions and recommendations. Later functionality introduced support for images and uploaded medical documents. Unlike MedixClarity, its primary focus is symptom-based assessment rather than structured extraction and interpretation of laboratory reports.

Regina Maria Virtual Assistant, launched in July 2022, uses conversational AI developed by DRUID together with Infermedica technology to guide patients toward an appropriate medical specialty based on reported symptoms [22]. The assistant operates within the Regina Maria healthcare ecosystem [3]. Its main objective is symptom assessment and patient routing rather than automated laboratory-document interpretation.

Cindi is a medical document interpretation platform created in Romania that processes uploaded medical documents using multiple AI models [19]. Among the Romanian applications analyzed, Cindi is conceptually close to MedixClarity because both systems accept medical documents and generate patient-oriented interpretations. MedixClarity differs through its explicit OCR-to-parser pipeline, post-extraction validation mechanisms, lexical RAG component, and self-hosted AI inference approach.

Ada Health is an internationally deployed symptom-assessment application with millions of completed health assessments [1]. Clinical evaluation studies have investigated its diagnostic recommendation performance [10]. Ada temporarily withdrew Romanian-language support and availability in Romania in early 2026, but Romanian support was subsequently restored in 2026 [2]. Unlike MedixClarity, Ada primarily focuses on symptom-based health assessment rather than extracting and interpreting laboratory values from uploaded reports.

Medicus AI Smart Reports is an international platform specifically designed to transform laboratory results into patient-oriented explanations [15]. The system combines medical content with LLM-generated narratives and provides conversational functionality for follow-up questions about laboratory results. Compared with MedixClarity, it addresses a similar goal of improving patient understanding

of laboratory results, while MedixClarity emphasizes an explicit OCR-to-parser pipeline, lexical RAG retrieval, and post-extraction validation mechanisms.

Recent research has also investigated the use of large language models for explaining laboratory test results to non-expert users. He et al. evaluated general-purpose and medically oriented LLMs on patient questions related to laboratory results and found that generated answers can be useful while still exhibiting limitations such as insufficient clinical context, incorrect statements, and limited personalization [12]. These findings motivate additional mechanisms such as retrieval augmentation, structured prompting, and response validation when applying LLMs to medical result interpretation.

Table 1 summarizes the main characteristics of the analyzed applications compared with MedixClarity.

The comparative analysis highlights several differences between MedixClarity and the existing solutions. Unlike Synevo Decoder and Regina Maria Virtual Assistant, which are integrated within specific healthcare provider ecosystems, MedixClarity is designed to accept reports regardless of the originating laboratory. Unlike SymptoChat and Ada Health, which primarily focus on symptom-based assessment, MedixClarity processes laboratory reports after medical investigations. Cindi and Medicus AI Smart Reports are conceptually closer to MedixClarity because they also provide patient-oriented interpretation of medical documents or laboratory results. MedixClarity differentiates itself through its explicit OCR-to-parser processing pipeline, post-extraction validation mechanisms, lexical RAG system with over 230 medical terms and more than 1500 aliases, and self-hosted LLaMA 3.1:8b inference used for interpretation and interactive follow-up.

PROPOSED SOLUTION

System Architecture

MedixClarity is built on a microservices architecture consisting of five independent components with clearly defined responsibilities: the FastAPI Backend, OCR Service, Parser Service, React Frontend, and PostgreSQL database. This architectural choice enables independent development, testing, and deployment of individual components [13, 24].

The platform is deployed across two cloud infrastructures under the control of the application: a Hetzner VPS hosts the persistent services (backend, parser, frontend, PostgreSQL database, and Nginx reverse proxy), while an on-demand RunPod GPU instance hosts computationally intensive OCR and LLM inference services. Ollama runs LLaMA 3.1:8b [11] on an NVIDIA RTX 4090 with 24 GB VRAM.

Although the underlying compute resources are provided by third-party infrastructure providers, MedixClarity does not transmit medical reports or extracted medical content to third-party AI inference APIs. Instead, the OCR and language models are self-hosted within compute instances provisioned and controlled by the application. Communication between deployed services follows a REST-based architecture, with retry logic using progressive backoff intervals of 5 and 10 seconds to handle temporary network interruptions.

The evaluated prototype relies on REST-based communication between distributed services. Transport-layer security should be enforced through HTTPS/TLS in production deployments, particularly for communication involving sensitive medical data across external cloud infrastructure. This

Application	Functionality	AI Technology	Medical Processing	Doc.	Processing / Deployment	Romania Availability
Synevo De-coder	Analysis interpretation via virtual consultation	Human physicians	No		Healthcare-provider ecosystem	Yes (Synevo patients)
SymptoChat	Symptom pre-diagnosis	Proprietary generative AI	Partial		Service-dependent	Yes
Regina Maria	Symptom assessment	Conversational AI / Infermedica	No		Healthcare-provider ecosystem	Yes (network patients)
Cindi	Medical document interpretation	Multiple AI models	Yes		AI-based document processing	Yes
Ada Health	Symptom assessment	Proprietary AI	No		Service-based deployment	Yes
Medicus AI Smart Reports	Laboratory-result interpretation	LLM-based explanations	Yes		Deployment-dependent	International
MedixClarity	Automated medical-report interpretation	Self-hosted LLaMA 3.1:8b	Yes (PDF and images)		Application-controlled cloud instances	Yes

Table 1: Comparison between analyzed applications and MedixClarity.

requirement forms part of the production hardening of the platform and complements the application-level privacy mechanisms described later in this section.

End-to-End Processing Pipeline

The data flow follows an automatic pipeline triggered by the user's upload action. The authenticated user submits a medical file in PDF or image format through the React interface. The FastAPI backend processes the file transiently during the analysis pipeline without retaining a permanent copy of the original document, reducing unnecessary storage of sensitive medical data [7]. A database record is created with *processing* status and the processing pipeline is launched without blocking the initial HTTP response.

The pipeline consists of three sequential stages. In the first stage, the backend transmits the file to the OCR service, which converts PDF documents page by page using PyMuPDF [21] at 200 DPI, applies conditional image preprocessing, and extracts text using PaddleOCR PP-OCRv4 [5, 18].

In the second stage, the extracted text is sent to the Parser service, which forwards it to Ollama [17] with a specialized extraction prompt containing strict rules covering numerical-value conversion, reference-interval handling, status-determination logic, and mandatory JSON output. Few-shot examples guide the expected output structure.

In the third stage, the backend constructs the final interpretation prompt using patient context and retrieved medical knowledge [8], sends the request to the self-hosted language model, and stores the generated interpretation.

OCR Module

The OCR service accepts JPEG, PNG, and PDF files up to 10 MB and 20 pages, with validation performed on both MIME type and magic bytes. PDF files are processed page by page using PyMuPDF [21] at 200 DPI. Conditional preprocessing is applied only when image quality is insufficient, evaluated using low contrast, sparse-text, and low-resolution criteria. When triggered, preprocessing applies Contrast Limited Adaptive Histogram Equalization (CLAHE), bicubic up-scaling for low-resolution images, and RGB conversion for PaddleOCR compatibility. The OCR engine uses PaddleOCR PP-OCRv4 [5, 18] with automatic text-orientation detection.

Parser Service and Anti-Hallucination Validation

The Parser service receives raw OCR text and returns structured JSON containing extracted medical tests with values, units, reference intervals, and status. The extraction prompt defines strict rules and contains few-shot examples. Input text is limited to the available model context to reduce the risk of incomplete structured responses. Generation parameters use a low temperature to favor deterministic extraction.

Parser responses undergo a six-step validation pipeline: (1) reference-interval correction for common errors involving <X and >X intervals; (2) numeric conversion of values and intervals; (3) independent status recalculation using corrected numerical values; (4) an anti-hallucination check verifying that extracted numeric values occur in the OCR text; (5) unit normalization for missing or inconsistent values; and (6) recovery of malformed JSON responses where possible.

This validation mechanism reduces reliance on model-generated status labels and provides an additional consistency layer between OCR output and structured medical data.

Lexical RAG System

The retrieval-augmented generation (RAG) knowledge base contains over 230 medical terms with more than 1500 aliases, organized across 14 clinical domains including hematology, biochemistry, lipid panel, liver and kidney function, endocrinology, tumor markers, and urinalysis [8, 9, 23].

Each entry may contain a canonical term, aliases, category, description, reference information, interpretation information, associated symptoms, and recommendations.

The lexical scoring algorithm evaluates candidate medical terms through several matching mechanisms, including exact keyword matching, aliases, patient test-name matching, and fuzzy string similarity. Terms below a minimum relevance threshold are filtered and the highest-scoring terms are injected into the interpretation prompt. In the evaluated implementation, the retrieval mechanism achieved an average latency of 10.93 ms using pre-computed indexes initialized at application startup.

The knowledge base was developed through a semi-automated process combining AI-assisted information extraction and manual validation against the consulted medical reference sources.

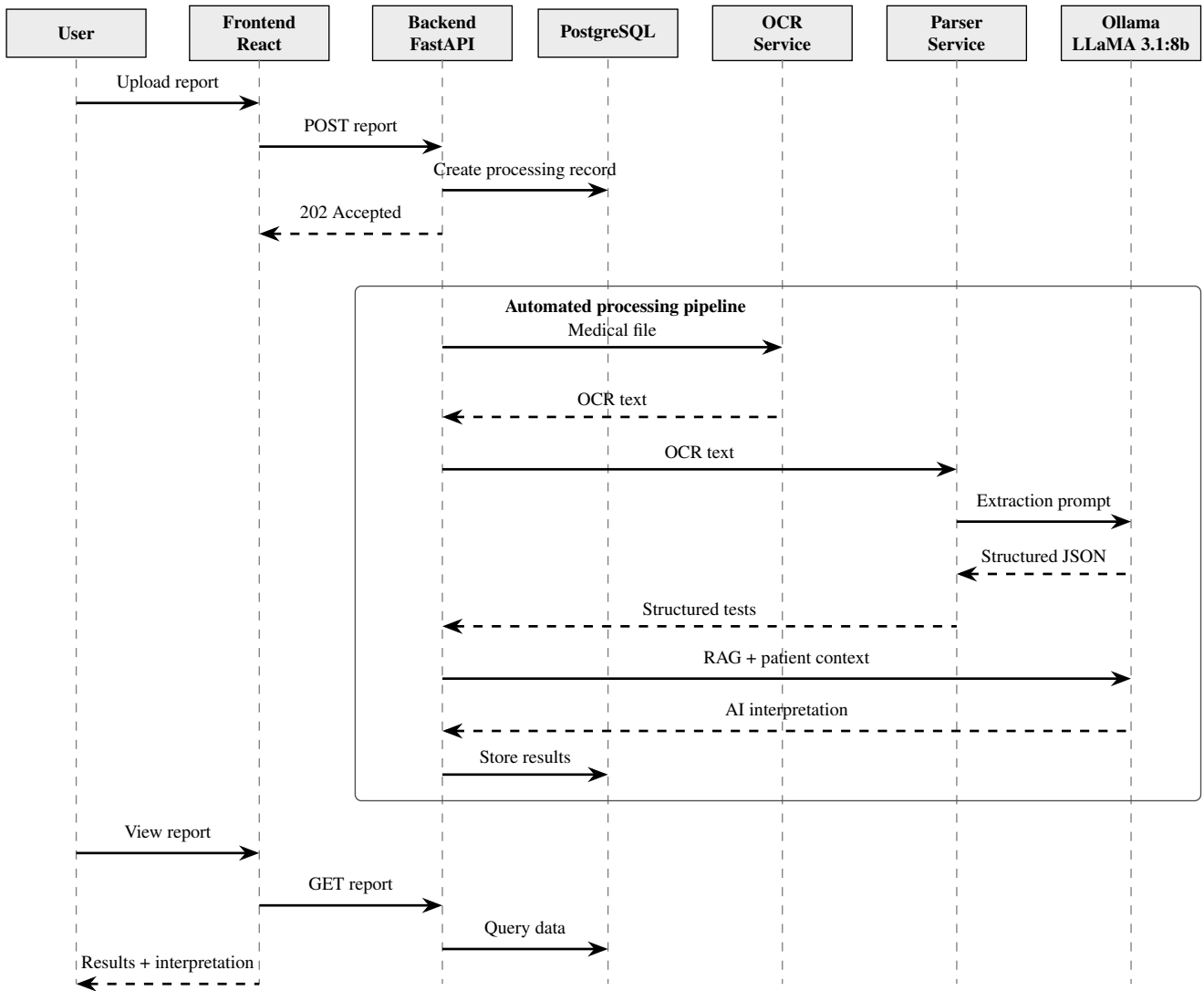


Figure 1: Sequence diagram of the MedixClarity processing pipeline, from report upload to OCR, structured parsing, AI interpretation, storage, and result retrieval.

Privacy and GDPR Considerations

MedixClarity was designed according to the data protection by design and data protection by default principles described in Article 25 of the GDPR [6, 7]. In practical terms, these principles require data-protection safeguards to be considered from the design stage of a system and personal-data processing to be limited, by default, to what is necessary for the intended purpose.

Several architectural decisions support these principles. Medical reports and extracted medical content are not transmitted to external AI inference APIs; instead, OCR and language-model inference are executed by services self-hosted on compute instances controlled by the application. Uploaded documents are processed transiently during the analysis pipeline rather than retained as permanent copies of the original files.

Processed medical information is stored in the application’s PostgreSQL database, while sensitive fields, including OCR text, AI-generated interpretations, and chat messages, are

encrypted at application level using Fernet authenticated encryption. Users can permanently delete their reports and associated data through a hard-delete operation, supporting the implementation of data-erasure functionality under the GDPR [7].

These technical measures reduce unnecessary data exposure and support GDPR-oriented privacy principles; however, they should not be interpreted as establishing GDPR compliance by themselves. Full compliance also depends on organizational and legal measures, including the lawful basis for processing, controller and processor responsibilities, retention policies, access-control procedures, and other obligations applicable to a production deployment.

EXPERIMENTS AND EVALUATION

The evaluation presented in this section is exploratory and is intended to characterize the behavior of the implemented pipeline rather than to establish general clinical or diagnostic accuracy. The technical evaluation is based on two real medical reports with different laboratory content and complexity, complemented by an exploratory usability study with

12 participants. Consequently, the reported OCR confidence, processing latency, extraction behavior, and interpretation quality should be interpreted as observations from these evaluated cases rather than as general performance estimates for arbitrary medical reports.

End-to-End Use-Case Scenarios

Three representative end-to-end scenarios illustrate the platform's functionality and cover the complete user-system interaction flow.

Scenario 1: Hematological and biochemical PDF report. An authenticated user uploads a three-page PDF report containing hematology, biochemistry, immunology, and urine-summary sections. The system extracts 43 tests, of which 36 are classified as normal and 7 as abnormal. The OCR stage extracted 7896 characters across 337 lines with a reported confidence score of 95.7% in 18.98 s. The Parser structured the tests in 55.44 s, while the interpretation was generated in 8.90 s, resulting in an initial total processing time of 83.32 s. The resulting dashboard is shown in Figure 2.

Scenario 2: Hematological PDF report. A second three-page PDF containing hematology and urine-summary results was processed. The OCR service extracted 5362 characters across 239 lines with a reported confidence score of 95.9% in 12.43 s. The Parser identified 30 tests in 35.84 s and the AI interpretation was generated in 14.68 s, resulting in an initial total processing time of 62.94 s.

Scenario 3: Interactive chat for clarifications. After viewing the interpretation generated for Report 1, the user can ask specific questions through the interactive chat. For each question, the platform builds a personalized prompt integrating recent conversation history, extracted report tests, relevant RAG terms obtained through lexical matching, and patient-profile information such as age and biological sex. Responses are streamed via Server-Sent Events (SSE) [14, 26], while conversation history is persisted in PostgreSQL [20] for cross-session continuity.

OCR Module Evaluation

The OCR module was evaluated on the two medical reports using PaddleOCR PP-OCRv4 [5, 18].

For Report 1, the service extracted 7896 characters across 337 lines with a confidence score of 95.7% in 18.98 s. The extracted text preserved the laboratory-value sections, including test names, numerical values, and reference intervals, without significant observed omissions in those sections.

For Report 2, the service extracted 5362 characters across 239 lines with a confidence score of 95.9% in 12.43 s. Recognition errors were observed mainly in the graphical header region, for example "ANALIREMEDELE" instead of "ANALIZE MEDICALE". These errors did not affect the laboratory-value sections in the evaluated document.

RAG System Evaluation

RAG in automatic interpretation. For Report 1, containing 43 tests, the RAG system retrieved four relevant medical terms: Eosinophils, Basophils, Urine Summary Examination, and Urea. The retrieved context included terms related to several abnormal values present in the report.

For Report 2, containing 30 tests, the system retrieved four hematological terms: Hemoglobin, Erythrocytes, Leukocytes, and Platelets.

Stage	R1 Init.	R1 Opt.	R2 Init.	R2 Opt.
OCR	18.98 s	10.21 s	12.43 s	6.55 s
Parser	55.44 s	26.13 s	35.84 s	24.49 s
AI Interpret	8.90 s	7.48 s	14.68 s	7.46 s
Total	83.32 s	43.82 s	62.94 s	38.50 s
Tests	43	43	30	30
Improvement	—	−47%	—	−39%

Table 2: End-to-end processing time before and after optimization.

RAG in interactive chat. Evaluation using four consecutive questions on Report 1 revealed variability depending on question formulation. For a generic query asking about abnormal values, the retrieved terms were not sufficiently aligned with all abnormal parameters, contributing to a partial answer identifying only 2 of 7 abnormal values. For explicit terminology queries, such as a question about LDL cholesterol, retrieval was more accurate. These observations illustrate the limitations of lexical retrieval for broad free-form questions [8, 23].

AI Interpretation Quality

Report 1. The prompt transmitted to LLaMA 3.1:8b [11] incorporated patient demographics, extracted medical tests, and retrieved RAG terms. The generated interpretation identified elevated eosinophils and basophils as relevant findings. However, it omitted the elevated Urea value and did not address some biochemical parameters for which reference intervals had not been completely extracted by the parser.

Report 2. The prompt contained demographic context corresponding to age 50 and male sex rather than the actual document context of age 40 and female sex, because the platform retrieved demographics from the user account profile rather than from document metadata. The generated interpretation stated that all parameters were within normal limits.

This case illustrates an important safety limitation: when demographic information used by the interpretation pipeline differs from the patient represented in the uploaded document, demographic-dependent reference ranges may be interpreted incorrectly. The failure therefore propagates from contextual data selection into final AI interpretation.

Performance and End-to-End Latency

Performance was evaluated by measuring processing times for each pipeline stage across the two reports, with measurements stored in PostgreSQL [20].

The Parser stage represented the largest proportion of the initial processing time for both evaluated reports. Post-optimization times were reduced through two main changes: configuring Ollama [17] to keep the model loaded in GPU memory and optimizing inter-service communication. In the two evaluated cases, total optimized processing time was reduced to approximately 38–44 seconds.

Usability Study

The exploratory usability study involved 12 participants from three broad user categories: medical university students, older users without advanced technical experience, and young adults with average digital experience.

The participant group included 5 people aged 20–30, 2 aged 31–40, 2 aged 41–50, and 3 aged over 50. Ten of the 12

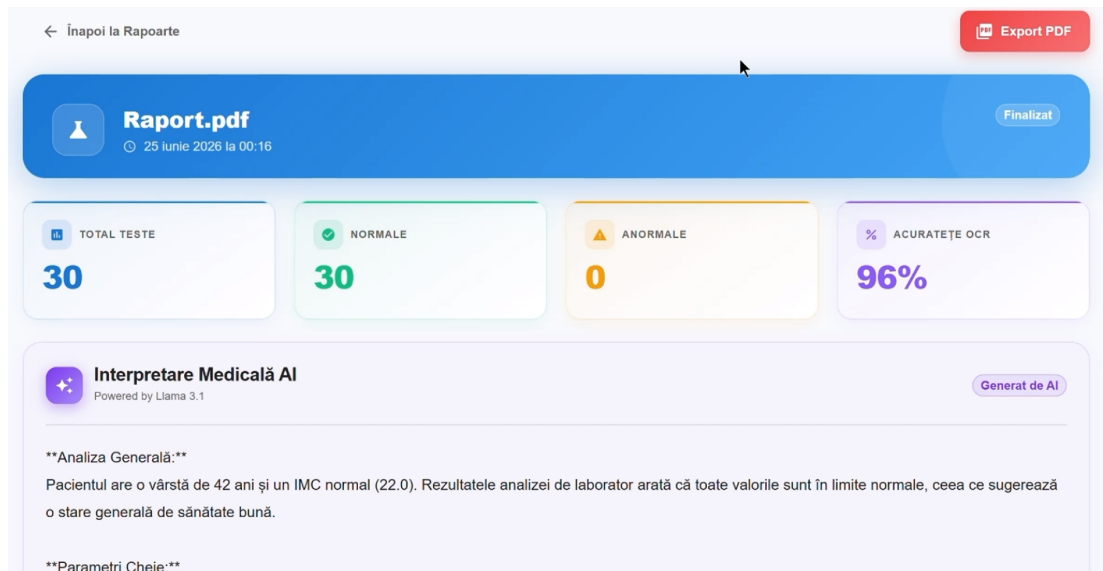


Figure 2: MedixClarity report results dashboard showing extracted laboratory values and the generated AI interpretation.

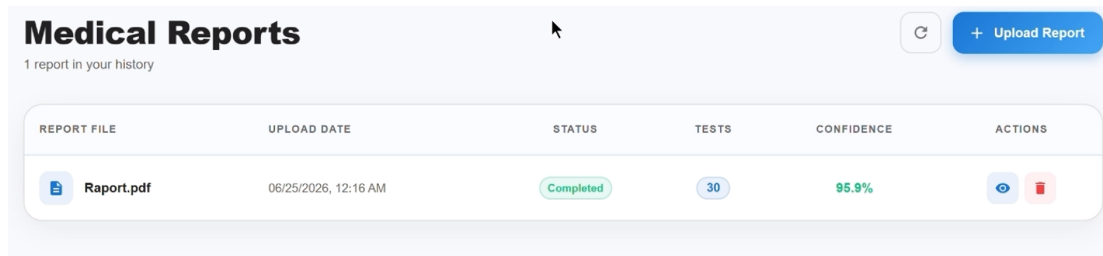


Figure 3: Medical report history view showing processed reports and OCR confidence information.

Criterion	Mean
Ease of registration/profile completion	4.83
Ease of medical-document upload	4.92
Clarity of AI interpretation	4.75
Utility of interactive chat	4.83
Trust in local data processing	5.00
Overall mean	4.87

Table 3: Usability results on a 1–5 Likert scale for 12 participants.

participants had not previously used a digital application for medical-test interpretation.

Participants completed three tasks: (1) registration and medical-profile completion; (2) uploading a medical PDF and reading the generated interpretation; and (3) accessing the interactive chat and asking at least one question.

All 12 participants completed the three tasks in the evaluated sessions.

The maximum mean score of 5.00 for trust in local data processing indicates a positive perception of the privacy-oriented approach among participants in this exploratory study. Eleven of the 12 participants stated that they would use MedixClarity regularly. Suggested improvements included voice output for interpretations, more detailed explanations for individual tests, and direct document-photo capture.

Because of the limited sample size, usability results were analyzed at aggregate level rather than statistically compared across participant subgroups. A larger follow-up study should

The 'Summary' screen displays patient data for review before saving. It includes sections for Personal data, Physical data, Lifestyle, and Medical history.

Personal data (Edit): BIRTH YEAR: 1984, BIOLOGICAL GENDER: Female

Physical data (Edit): HEIGHT (CM): 165 cm, WEIGHT (KG): 60 kg, BMI: 22.04 (Normal)

Lifestyle (Edit): SMOKING STATUS: Non-smoker

Medical history (Edit): No conditions selected

Buttons: Back, Confirm and save

Figure 4: Patient profile onboarding summary screen.

include more participants from each user category and report usability measures separately for medical students, older adults, and users with different levels of digital experience.

Identified Limitations and Error Categories

The evaluation revealed several categories of errors affecting different stages of the MedixClarity pipeline.

OCR-related errors. Recognition accuracy depends on document layout and scan quality. In Report 2, observed errors were localized mainly in the graphical header area, while laboratory values remained correctly recognized in the evaluated document.

Parser and reference-range errors. The Parser may incorrectly handle conditional reference intervals, such as age-dependent ranges, and may misinterpret concatenated unit notation. One observed example involved a platelet value being interpreted as 32,600 instead of 326,000 /uL. Such errors may propagate into status classification and subsequent interpretation.

Demographic-context errors. The current pipeline retrieves demographic information from the user's profile rather than extracting and reconciling it with demographic information present in the uploaded document. Report 2 demonstrated the safety implications of this design: interpretation was generated using a profile corresponding to a 50-year-old male although the document represented a 40-year-old female.

A safer production-oriented design should extract demographic attributes from the medical document when available, compare them with the user profile, and explicitly detect conflicts. When a discrepancy is found, the system should request user confirmation before generating an interpretation or avoid demographic-dependent interpretation until the information has been validated.

RAG retrieval errors. The lexical RAG mechanism performs more consistently for explicit medical terminology than for broad or ambiguous free-form questions. Irrelevant or weakly related terms may occasionally be retrieved. This limitation motivates future investigation of semantic retrieval based on embeddings and vector similarity.

LLM interpretation errors. The language model may omit abnormal values, generate terminology errors, or provide incomplete interpretations when input context is ambiguous or partially incorrect. These observations motivate stronger post-generation validation and evaluation of domain-specialized medical language models.

Context-window and interaction limitations. Extended reports and long chat histories may approach the available model context window. In interactive chat mode, response quality may also vary as conversation history accumulates and retrieval results change between questions.

Overall, these failure modes show that errors can propagate through multiple stages of the pipeline, from OCR and structured extraction to retrieval and final interpretation. MedixClarity should therefore be considered an assistive interpretation system rather than a diagnostic tool, and medically significant conclusions should remain subject to professional clinical review.

CONCLUSIONS

MedixClarity addresses the need for accessible and privacy-oriented interpretation of medical laboratory reports for users without specialized medical training. The platform combines optical character recognition [18], self-hosted LLaMA 3.1:8b inference [11], structured parsing with post-extraction validation, and a lexical RAG system [8] to transform uploaded medical documents into structured results, patient-oriented explanations, and interactive follow-up responses.

In the two evaluated medical reports, the complete pipeline processed the documents end-to-end within approximately

38–83 seconds depending on configuration, while reported OCR confidence scores exceeded 95%. The exploratory usability study with 12 participants resulted in an overall mean Likert score of 4.87/5 and a 100% task-completion rate for the evaluated tasks. These results indicate that the implemented architecture is technically feasible and was positively perceived by the study participants, while the identified failure cases demonstrate that interpretation quality remains sensitive to OCR quality, structured extraction, demographic context, retrieval relevance, and LLM behavior.

Future work should prioritize evaluation on a larger and more diverse benchmark containing reports from multiple laboratories, document layouts, scan qualities, clinical domains, and demographic groups, with manually validated reference data for measuring OCR, extraction, and interpretation accuracy. A larger usability study should also include more participants from each user category and analyze results separately across different age groups, medical backgrounds, and levels of digital experience.

Further technical development includes detecting and reconciling demographic information between uploaded documents and user profiles before interpretation, strengthening post-generation safety validation, expanding the medical knowledge base, and replacing or complementing lexical retrieval with semantic RAG based on embeddings and vector search [9, 23]. Additional directions include evaluating medical-specialized open-source language models and developing a native mobile application.

ACKNOWLEDGMENTS

Generative AI tools were used to assist with the construction of the MedixClarity medical knowledge base and with identifying grammatical, spelling, and writing issues in the manuscript. All AI-assisted content and suggestions were reviewed by the authors, who remain fully responsible for the final content.

References

- [1] Ada Health. 2020. *Let Ada Connect You to a Symptom Checker* 24/7. <https://ada.com/ro/app/>
- [2] Ada Health. 2026. *Important Update Regarding Access to the Ada Application in Romania*. <https://ada.com/ro/editorial/important-update/>
- [3] F. Chreih. 2024. *AI in laborator si imagistica, o inovatie adusa de REGINA MARIA pe piata*. <https://www.forbes.ro/ai-in-laborator-si-imagistica-389434>
- [4] V. Constantinescu. 2024. *Trei romani au lansat o aplicatie de diagnosticare medicala ce foloseste AI*. <https://www.wall-street.ro/articol/Start-Up/311683/trei-romani-au-lansat-o-aplicatie-de-diagnosticare-medicala-ce-foloseste-ai.html>
- [5] Cheng Cui et al. 2025. *PaddleOCR 3.0 Technical Report*. *arXiv preprint arXiv:2507.05595* (2025). <https://arxiv.org/abs/2507.05595>
- [6] European Data Protection Board. 2019. *Guidelines 4/2019 on Article 25 Data Protection by Design and by Default*. Technical Report. European Data Protection Board. https://www.edpb.europa.eu/sites/default/files/files/file1/edpb_guidelines_201904_dataprotection_by_design_and_by_default_v2.0_en.pdf
- [7] European Parliament and Council of the European Union. 2016. *Regulation (EU) 2016/679 on the Protection of Natural Persons with Regard to the Processing of Personal Data (GDPR)*. Technical Report. EUR-Lex. <https://eur-lex.europa.eu/eli/reg/2016/679/oj>
- [8] Yunfan Gao et al. 2023. *Retrieval-Augmented Generation for Large Language Models: A Survey*. *arXiv preprint arXiv:2312.10997* (2023). <https://arxiv.org/abs/2312.10997>
- [9] Omid Kohandel Gargari and Gholamreza Habibi. 2025. *Enhancing Medical AI with Retrieval-Augmented Generation: A Mini Narrative Review*. *Digital Health* 11 (2025), 1–7. doi:10.1177/20552076251337177
- [10] Stephen Gilbert, Matthew Fenech, Shubhanan Upadhyay, Paul Wicks, and Claire Novorol. 2021. *Quality of Condition Suggestions and Urgency Advice Provided by the Ada Symptom Assessment App Evaluated with Vignettes Optimised for Australia*. *Australian Journal of Primary Health* 27, 5 (2021), 377–381. doi:10.1071/PY21032
- [11] Aaron Grattafiori et al. 2024. *The Llama 3 Herd of Models*. *arXiv preprint arXiv:2407.21783* (2024). <https://arxiv.org/abs/2407.21783>

- [12] Zhe He, Balu Bhasuran, Qiao Jin, Shubo Tian, Karim Hanna, Cindy Shavor, Lisbeth García Arguello, Patrick Murray, and Zhiyong Lu. 2024. Quality of Answers of Generative Large Language Models Versus Peer Users for Interpreting Laboratory Test Results for Lay Patients: Evaluation Study. *Journal of Medical Internet Research* 26 (2024), e56655. doi:10.2196/56655
- [13] James Lewis and Martin Fowler. 2014. Microservices: A Definition of This New Architectural Term. *martinfowler.com* (2014). <https://martinfowler.com/articles/microservices.html>
- [14] MDN Web Docs. 2026. *Using Server-Sent Events*. https://developer.mozilla.org/en-US/docs/Web/API/Server-sent_events/Using_server-sent_events
- [15] Medicus AI. 2026. Smart Reports: AI-Powered Lab Result Interpretation. <https://www.medicus.ai/smart-reports>. Accessed: 2026-09-02.
- [16] Shervin Minaee et al. 2024. Large Language Models: A Survey. *arXiv preprint arXiv:2402.06196* (2024). <https://arxiv.org/abs/2402.06196>
- [17] Ollama. 2024. *API Introduction*. <https://docs.ollama.com/api/introduction>
- [18] PaddlePaddle. 2024. *PaddleOCR*. <https://github.com/PaddlePaddle/PaddleOCR>
- [19] D. Petruleac. 2025. *Un moldovean care a invins cancerul a creat o aplicatie care explica analizele medicale*. https://newsmaker.md/ro/un-moldovean-care-a-invins-cancerul_explica-analizele-medicale
- [20] PostgreSQL Global Development Group. 2026. *PostgreSQL: The World's Most Advanced Open Source Relational Database*. <https://www.postgresql.org/>
- [21] PyMuPDF Contributors. 2024. *PyMuPDF*. <https://github.com/pymupdf/PyMuPDF>
- [22] Regina Maria. 2022. *Regina Maria lanseaza, in premiera in Romania, primul Asistent Virtual care evalueaza online simptomele pacientilor*. <https://www.reginamaria.ro/stiri/regina-maria-lanseaza-premiera-romania-primul-asistent-virtual-care-evalueaza-online>
- [23] Chetan Sharma. 2025. Retrieval-Augmented Generation: A Comprehensive Survey of Architectures, Enhancements, and Robustness Frontiers. *arXiv preprint arXiv:2506.00054* (2025). <https://arxiv.org/abs/2506.00054>
- [24] Mert Söylemez, Bedir Tekinerdogan, and Ayça Kolukisa Tarhan. 2022. Challenges and Solution Directions of Microservice Architectures: A Systematic Literature Review. *Applied Sciences* 12, 11 (2022), 5507. doi:10.3390/app12115507
- [25] Synevo Romania. 2021. *Synevo Romania a lansat impreuna cu ATLAS primul serviciu de consultatii virtuale pentru interpretarea rezultatelor analizelor medicale*. <https://www.synevo.ro/serviciu-consultatii-virtuale-analizelor-medicale/>
- [26] WHATWG. 2026. *Server-Sent Events*. <https://html.spec.whatwg.org/multipage/server-sent-events.html>
- [27] Wayne Xin Zhao et al. 2023. A Survey of Large Language Models. *arXiv preprint arXiv:2303.18223* (2023). <https://arxiv.org/abs/2303.18223>