



iJRASET

International Journal For Research in
Applied Science and Engineering Technology



INTERNATIONAL JOURNAL FOR RESEARCH

IN APPLIED SCIENCE & ENGINEERING TECHNOLOGY

Volume: 14 **Issue:** X **Month of publication:** October 2026

DOI: <https://doi.org/10.22214/ijraset.2026.85133>

www.ijraset.com

Call: ☎ 08813907089

E-mail ID: ijraset@gmail.com

Vakilo AI - Legal Assistant Platform

Yogesh Bisane¹, Roshni Sharma², Samanta Narde³, Prachi Pandhurkar⁴, Aumkar Sarode⁵, Mr. Ashish Trivedi⁶

G.H. Rasoni University Amaravati / CSE Department, Nagpur, India

Abstract: Many people struggle to get legal help at the very start, because they cannot easily put their problem into legal terms or tell which lawyer would suit it. Vakilo AI is a web platform we built to guide that first stage: it walks a user through case intake and puts them in touch with verified legal professionals. The user describes the problem and chooses a category and an urgency level. An intake module powered by Gemini then poses ten follow-up questions and, from the answers, drafts a structured case summary with suggested next steps. The platform also proposes lawyers who may fit the case, and each of them can look over the request before accepting or declining it. Once a request is accepted, the two sides can chat privately, hold video consultations, book appointments, share case documents, and settle payments through Razorpay. A lawyer must pass administrative verification of their profile before being allowed to accept any case. The system is built from a Python FastAPI backend, a React front end, PostgreSQL and Supabase for data services, Socket.IO for messaging, and WebRTC for video. Vakilo AI is meant to bring order to the early stage of seeking legal help; what the AI generates is there to assist a qualified lawyer's advice, not to stand in for it.

Keywords: Legal technology, AI-assisted case intake, lawyer recommendation, online consultation, case management, real-time communication

I. INTRODUCTION

Whether someone gets justice often depends on whether they can find and reach the right legal professional, and that first step is harder than it sounds. A person facing a property dispute, a family matter, a criminal allegation, or a consumer complaint may not know which branch of law applies, what a lawyer will want to know, or how quickly they need to act. Referrals from friends and manual searches do help, yet neither makes it easy to compare lawyers' expertise or to keep the paperwork and conversations of a case in order. Law-firm websites and directories usually give contact details only, so the client ends up juggling separate tools for booking appointments, messaging, sending documents, and paying fees.

Advances in artificial intelligence and web development make it possible to bring some structure to this stage. Language models can gather information in a consistent format, and a web platform can keep records and communication together in a single workflow. Legal work, however, leaves little room for error: missing details or a wrong AI-generated answer can cause real harm, so such tools need to be used with care. In this paper we present Vakilo AI, a web platform that takes clients through a structured intake, generates an AI-assisted case summary, and recommends verified lawyers on the basis of the information submitted. It further provides client-lawyer messaging, video consultations, appointment scheduling, document sharing, payments, and status notifications. Our aim is to make the search for legal help better organized and easier to access, while leaving legal judgment to professionals and never presenting AI output as a replacement for it.

II. LITERATURE REVIEW

Research on AI in legal services spans a few connected areas: platforms that bring clients and lawyers together, language processing for legal documents, retrieval of relevant cases, and the dependability of machine-generated answers. Together they form the background for Vakilo AI, and they also show why a convenient workflow should not be confused with trustworthy legal information.

On the platform side, Avareddy et al. [1] present Legal Assist, a system that combines lawyer discovery, consultations, case management, document handling, and a legal-awareness chatbot. Gathering these everyday tasks in one interface is close to the workflow we propose, although the authors describe the chatbot's guidance as fairly general. Ariai et al. [3] survey natural language processing in the legal domain and weigh its promise against lasting obstacles such as accuracy, explainability, bias, robustness, scarce data, and the intricacy of legal reasoning. Chalkidis et al. [8] compare strategies for adapting BERT to legal text and release the LEGAL-BERT models. Because those models were trained mostly on European, UK, and US material, their performance on Indian legal text cannot be taken for granted without testing.

Finding the right legal material is a research problem in its own right. Arora et al. [2] found that pairing BM25 keyword search with topic information retrieves precedents better than keywords alone; their BERT-based classification of rhetorical roles in judgments, on the other hand, left room for improvement.

Feng et al. [5] review methods, datasets, and recent developments in legal case retrieval and observe that neural retrieval has advanced without making legal relevance any easier to define. Lewis et al. [9] proposed retrieval-augmented generation (RAG), in which a text generator is paired with an external searchable index. In principle this lets answers be tied to retrieved sources and lets the knowledge base be updated, though retrieving a source does not make the generated answer correct.

Reliability matters most when AI is asked to answer legal questions. Martinez-Gil [4] argues that legal question-answering systems have to cope with legal meaning and ambiguity, explain their outputs, and draw on trustworthy sources. Magesh et al. [10] tested commercial legal research tools built on RAG and found hallucinated content in some of the answers they examined. Their results suggest that retrieval lowers certain risks without removing the need for verification and human judgment.

Work on Indian legal data shows the difficulty clearly. Malik et al. [6] built the Indian Legal Documents Corpus (ILDC) from roughly 35,000 Supreme Court cases and used it to study judgment prediction together with explanations. Their best model reached 78% accuracy against 94% for human legal experts, and its explanations differed from those the experts gave. In a later study, Malik et al. [7] defined 13 rhetorical-role labels for segmenting Indian legal documents and compared a multitask model with baseline methods. Taken together, these studies caution against assuming that tools built for other jurisdictions will work as well on Indian judgments, which tend to be long and written in specialized language.

Reading across this literature, we see a practical gap. Platform studies pull legal-service tasks into one place, while legal NLP and retrieval research develop ways to process and search legal text. A client-facing system, though, also has to tie together intake, lawyer discovery, communication, and the activities around a case, and work on legal question answering warns against trusting generated answers by default. Vakilo AI tackles the workflow side of this gap through guided intake, AI-assisted summaries, lawyer recommendations, and collaboration features. In its present form it should be seen as a tool for organizing a case and reaching a lawyer, not as a validated engine for legal reasoning or legal research.

Authors and Findings

Ref.	Study / publication	Reported finding
[1]	Avareddy et al., IJERT, 2025	Brings lawyer discovery, consultation, case handling, and document management into one platform, with a chatbot that offers only general guidance.
[2]	Arora et al., FIRE 2020 (AILA)	Pairing BM25 with topic information improves precedent retrieval, while BERT-based rhetorical-role classification still has room to improve.
[3]	Ariai et al., ACM Comput. Surv., 2025	Legal NLP is highly promising, but accuracy, explainability, bias, and limited data remain obstacles.
[4]	Martinez-Gil, Comput. Sci. Rev., 2023	Dependable legal QA needs an understanding of legal semantics, explainable output, and trustworthy sources.
[5]	Feng et al., ACL 2024	Neural retrieval and new datasets have advanced legal case retrieval, yet judging legal relevance remains hard.
[6]	Malik et al., ACL-IJCNLP 2021	ILDC corpus of about 35,000 Supreme Court cases; best model reached 78% accuracy against 94% for human experts.
[7]	Malik et al., NLLP 2022	13 rhetorical-role labels, an annotated corpus, and a multitask segmentation model for Indian legal documents.
[8]	Chalkidis et al., Findings of EMNLP 2020	Compares ways of adapting BERT to law and releases the LEGAL-BERT model family.
[9]	Lewis et al., NeurIPS 2020	RAG couples a generator with a searchable index, helping with provenance and knowledge updates.
[10]	Magesh et al., J. Empir. Leg. Stud., 2025	Commercial legal RAG tools hallucinated in 17%–33% of tested queries, contrary to vendors' claims.

III. OBJECTIVES

The project set out to build Vakilo AI, an AI-assisted platform that helps users organize their legal concerns and get in touch with a suitable lawyer. Its specific objectives are to:

- 1) Offer a guided case-submission process that needs no prior legal knowledge.
- 2) Use AI to ask relevant follow-up questions and convert the answers into a structured case summary.
- 3) Suggest verified lawyers whose stated expertise matches the details of the case.
- 4) Enable private client-lawyer messaging and video consultations.
- 5) Handle document sharing, appointment scheduling, and payments within the same workflow.
- 6) Have an administrator review each lawyer profile before that lawyer can accept cases.
- 7) Keep clients and lawyers informed of updates on cases, messages, appointments, and documents.

A. Existing System

Clients usually find a lawyer through referrals, word of mouth, directories, or a lawyer's own website. These routes work to a degree, but they rarely reflect the type of case, how urgent it is, or the specialist experience it calls for. Many listing services stop at contact details and leave intake and follow-up to the client. Scheduling, document exchange, messaging, and payment therefore end up spread over unconnected channels, which makes it hard to keep one consistent record of the case. Where verification of profiles is weak or not clearly stated, clients may also have no easy way to tell whether a listed professional has been vetted.

B. Proposed System

Vakilo AI puts initial case intake and client-lawyer coordination into a single web application. The client describes the problem, picks a category, and marks how urgent it is. Using the Gemini API, the system generates ten follow-up questions suited to what was entered. Once the client has answered them, it produces a structured summary with suggested next steps and lists lawyers from the platform's verified profiles as recommendations. A lawyer can read an incoming request and either accept or decline it. On acceptance, the platform opens a private chat and supports video consultations, appointment scheduling, document sharing, and payment through Razorpay. Administrators check lawyer profiles before those lawyers may accept cases, and notifications keep both sides aware of activity on the platform.

C. Main System Features

- ☐ AI-assisted intake: The user supplies a case category, urgency level, and description, and the system follows up with ten questions to gather more context.
- ☐ Case summary: The AI turns the user's answers into a readable summary and lists possible next steps to discuss with a lawyer.
- ☐ Lawyer recommendations: Lawyers from the verified directory are suggested according to the case information and the content of their profiles.
- ☐ Lawyer verification: An administrator examines each submitted lawyer profile and its supporting documents before case acceptance is switched on.
- ☐ Request handling: A lawyer can look through the case details and choose to accept or decline the client's request.
- ☐ Private messaging: Accepting a request opens a client-lawyer chat space for communication about the case.
- ☐ Video consultations and appointments: Users can set up a consultation and pick a date and time.
- ☐ Document exchange: Clients and lawyers can share case-related files on the platform.
- ☐ Payment integration: Razorpay is built in so that clients can pay for legal services.
- ☐ Notifications: Updates on case status, messages, appointments, and documents are sent automatically.
- ☐ Role-specific dashboards: Clients, lawyers, and administrators each get their own interface, with tasks and permissions organized accordingly.

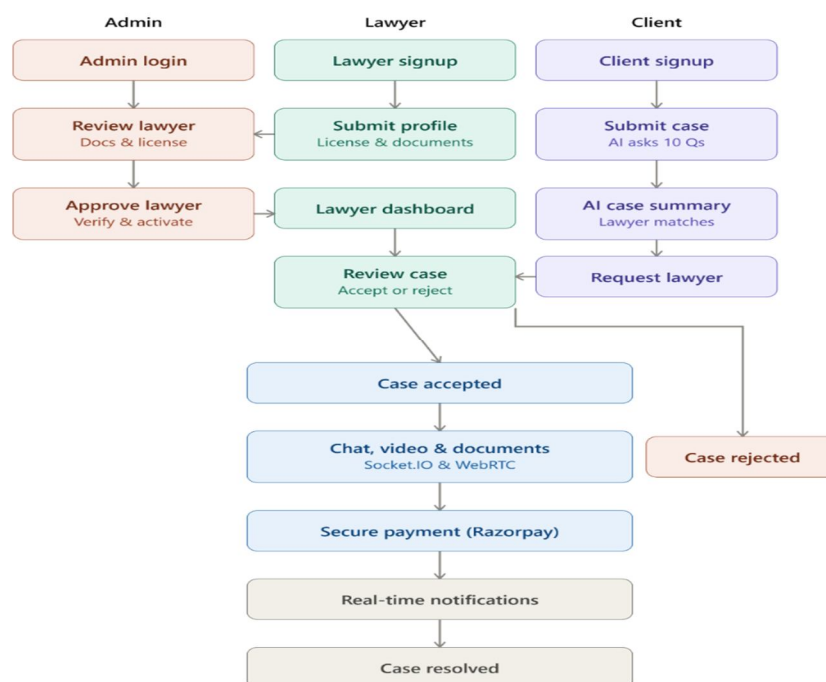
IV. METHODOLOGY

The system was developed in stages. We began by dividing the requirements into four modules: client, lawyer, administrator, and AI intake. The client workflow came next, allowing a user to submit a case description, category, and urgency level. From those details the Gemini API generates ten follow-up questions, and the user's answers are then used to prepare a case summary, suggested next steps, and lawyer recommendations drawn from the verified profiles in the system.

In the lawyer workflow, a professional submits a profile along with supporting documents. An administrator reviews this material and approves eligible profiles; only then can the lawyer accept cases. When a client requests a particular lawyer, that lawyer reviews the case and accepts or declines. Acceptance activates the client-lawyer communication workflow, with Socket.IO handling real-time messaging and WebRTC providing video consultation. The application additionally offers appointment scheduling, document sharing, Razorpay payments, and notifications about case activity.

The backend is written in Python with FastAPI and exposes REST APIs, while PostgreSQL and Supabase provide data storage and related services. The interface uses React and TypeScript, is styled with Tailwind CSS, and is built with Vite, with Node.js supporting the front-end development environment. These parts were combined into one application that has separate dashboards for clients, lawyers, and administrators. Our effort concentrated on getting the main workflows to run from start to finish. To judge how the system holds up in broader real-world use, formal evaluations of performance, security, usability, and the accuracy of legal answers would still be required.

Fig. A. Flow diagram of the Vakilo AI system



V. RESULTS AND DISCUSSION

A. Results

The implemented Vakilo AI application covers the main path from case submission to client-lawyer coordination. Users can register, sign in, and submit a case by describing the problem and choosing its category and urgency. The Gemini-powered intake module produces ten follow-up questions from those details. When the user has responded, the application shows a structured case summary and suggested next steps, together with recommendations drawn from the platform's verified lawyer profiles.

Lawyers can register and send in their profile information for administrative review, and only approved profiles are able to accept case requests. On receiving a request, a lawyer can examine the case information and then accept or decline. For accepted requests the application provides a private chat space. The implementation also includes real-time messaging through Socket.IO, WebRTC video consultation, appointment scheduling, document sharing, Razorpay payments, and notifications about case activity. Figures 1–15 present screenshots of the interface: the landing and sign-up pages, client and lawyer dashboards, AI intake and summary screens, lawyer recommendations, chat, document sharing, appointments, payment management, video consultation, the legal-section interface, and the administrator panel.

B. Discussion

The implementation shows that guided intake and case-management tools can sit together in a single application. The follow-up questions give users a structured way to supply information, and the generated summary makes it easier to go over that information before a consultation. Lawyer verification adds a checkpoint run by administrators, while built-in communication, scheduling, document, and payment features spare users from moving between separate services. Role-specific dashboards also keep the tasks of clients, lawyers, and administrators apart.

These observations concern the features built into the prototype. They do not prove that the platform is faster, more accurate, or more trustworthy than existing services. This paper reports no controlled user study, quantitative usability results, lawyer-matching accuracy, security testing, or independent evaluation of the AI-generated content. Such testing is the natural next step, especially since AI output can be incomplete or wrong. Generated summaries should be checked with a qualified legal professional, and the platform should not be regarded as a source of definitive legal advice.

1) Outputs

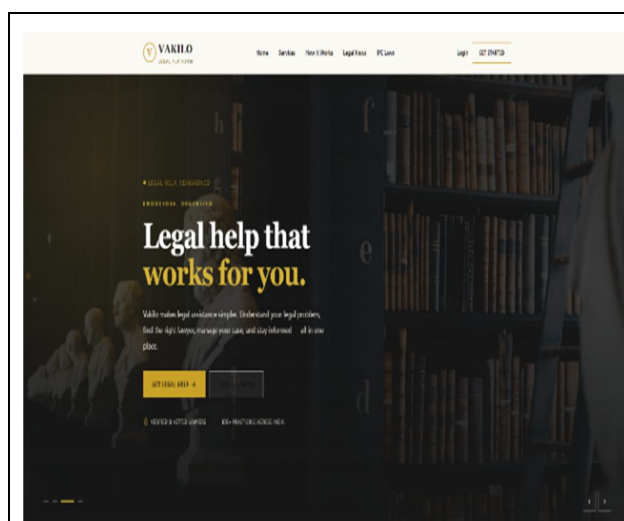


Fig. 1. Landing Page

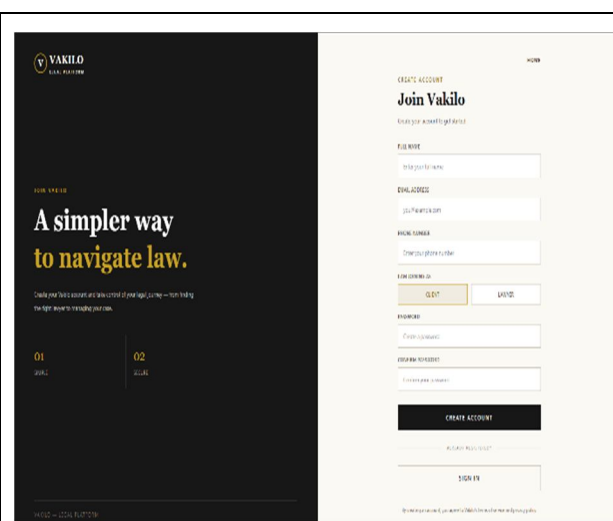


Fig. 2. SignUp Page

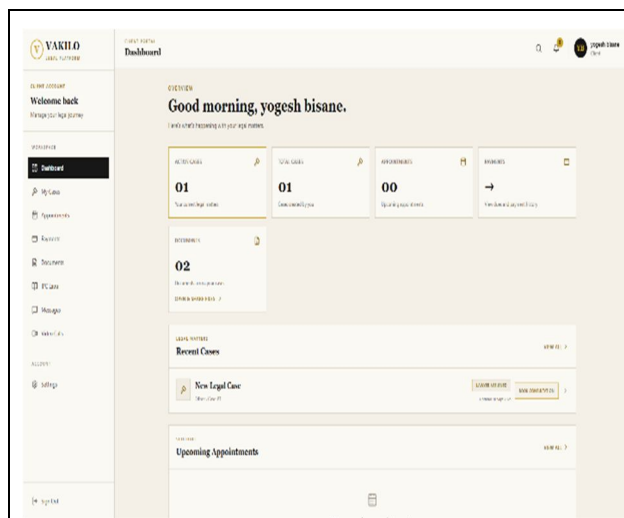


Fig. 3. Client Dashboard

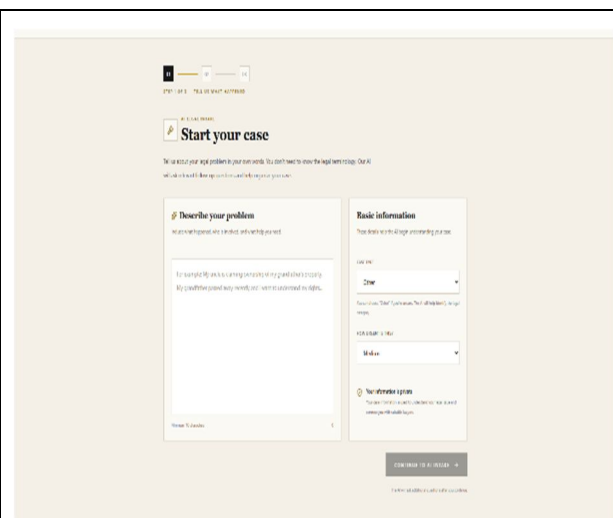


Fig. 4. Start case

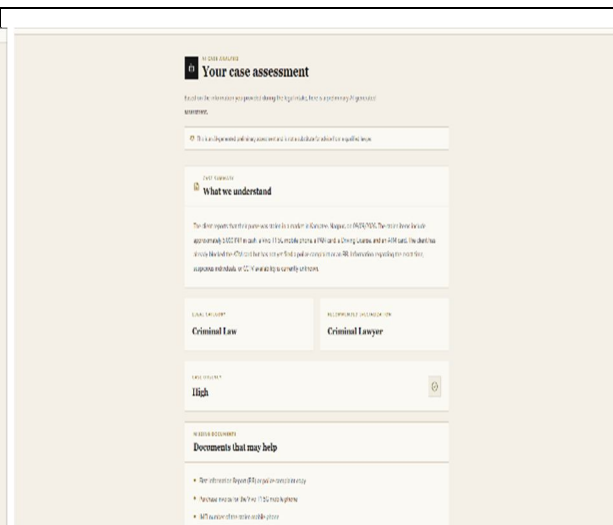


Fig. 6. AI Analyze Case Summary

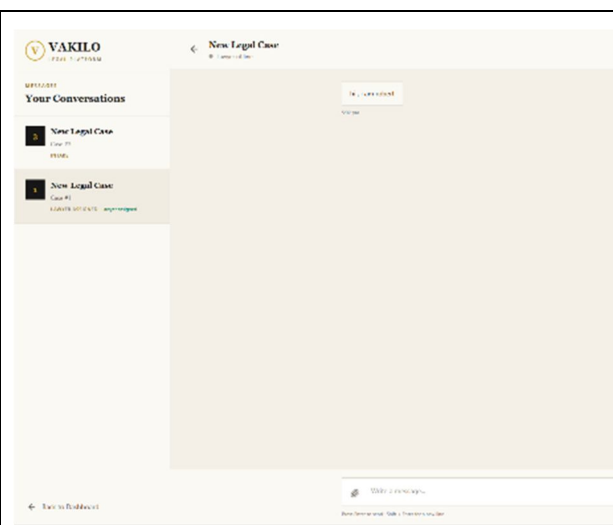


Fig. 8. Case Chat Room

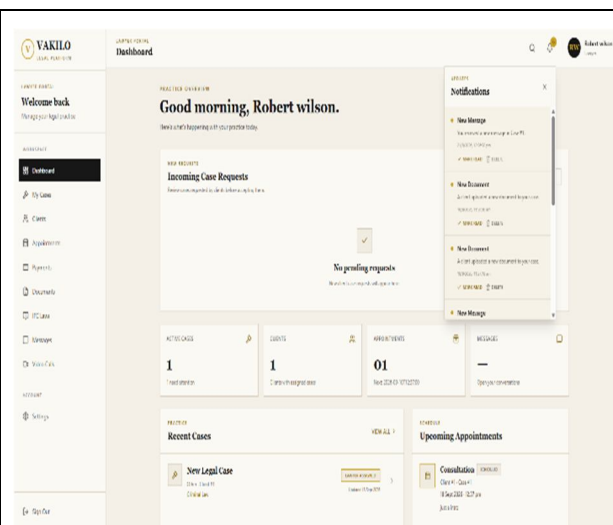


Fig. 10. Lawyer Dashboard

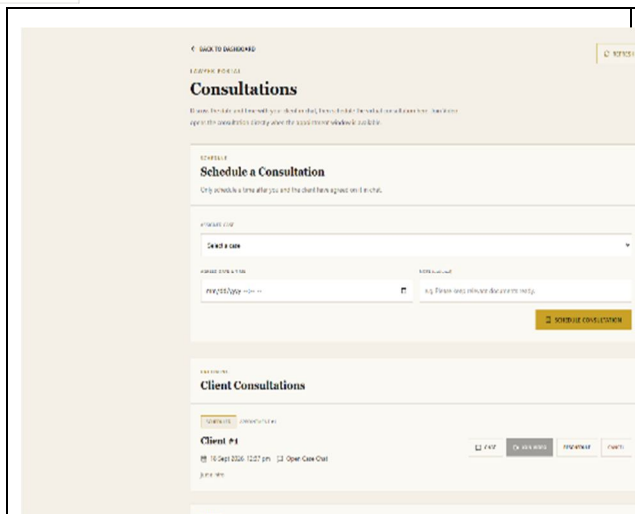


Fig. 11. Appointment Scheduling

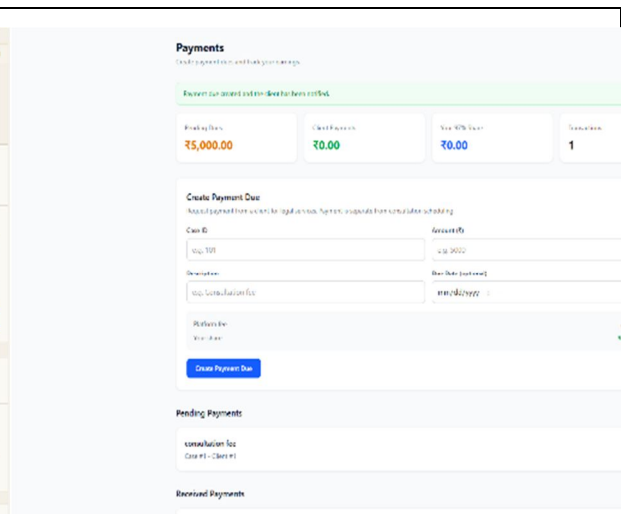


Fig. 12. Payment Managing Panel

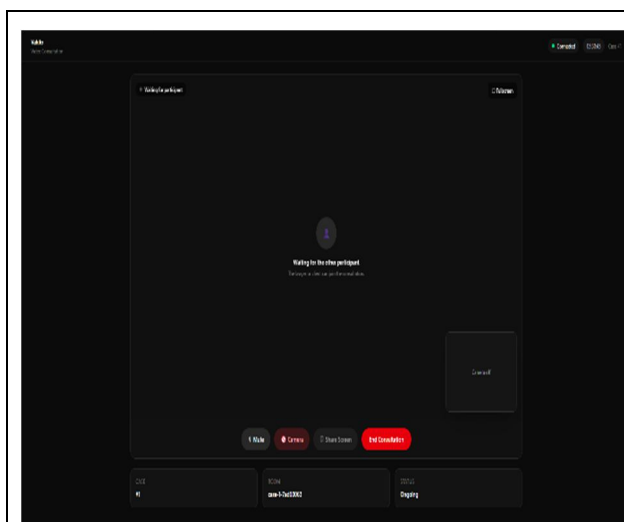


Fig. 13. Video conferencing Setup

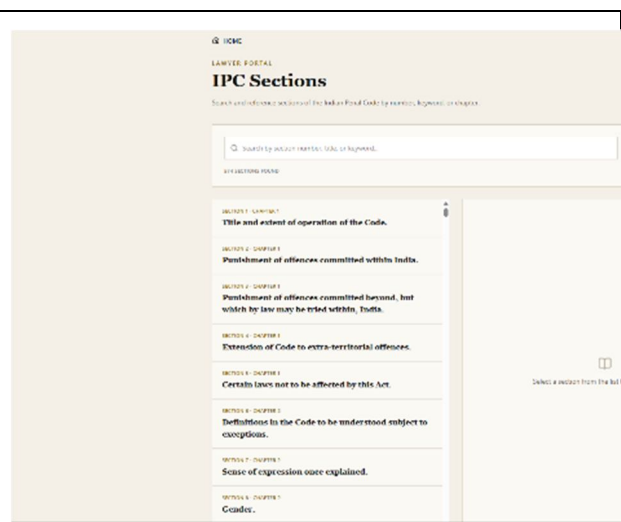


Fig. 14. IPC Section

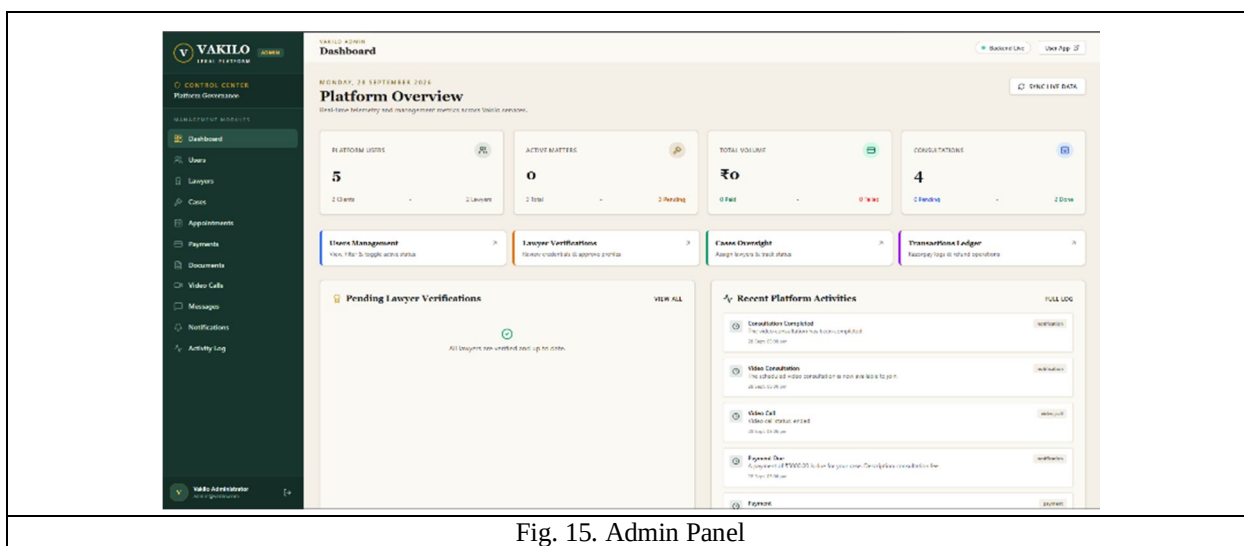


Fig. 15. Admin Panel

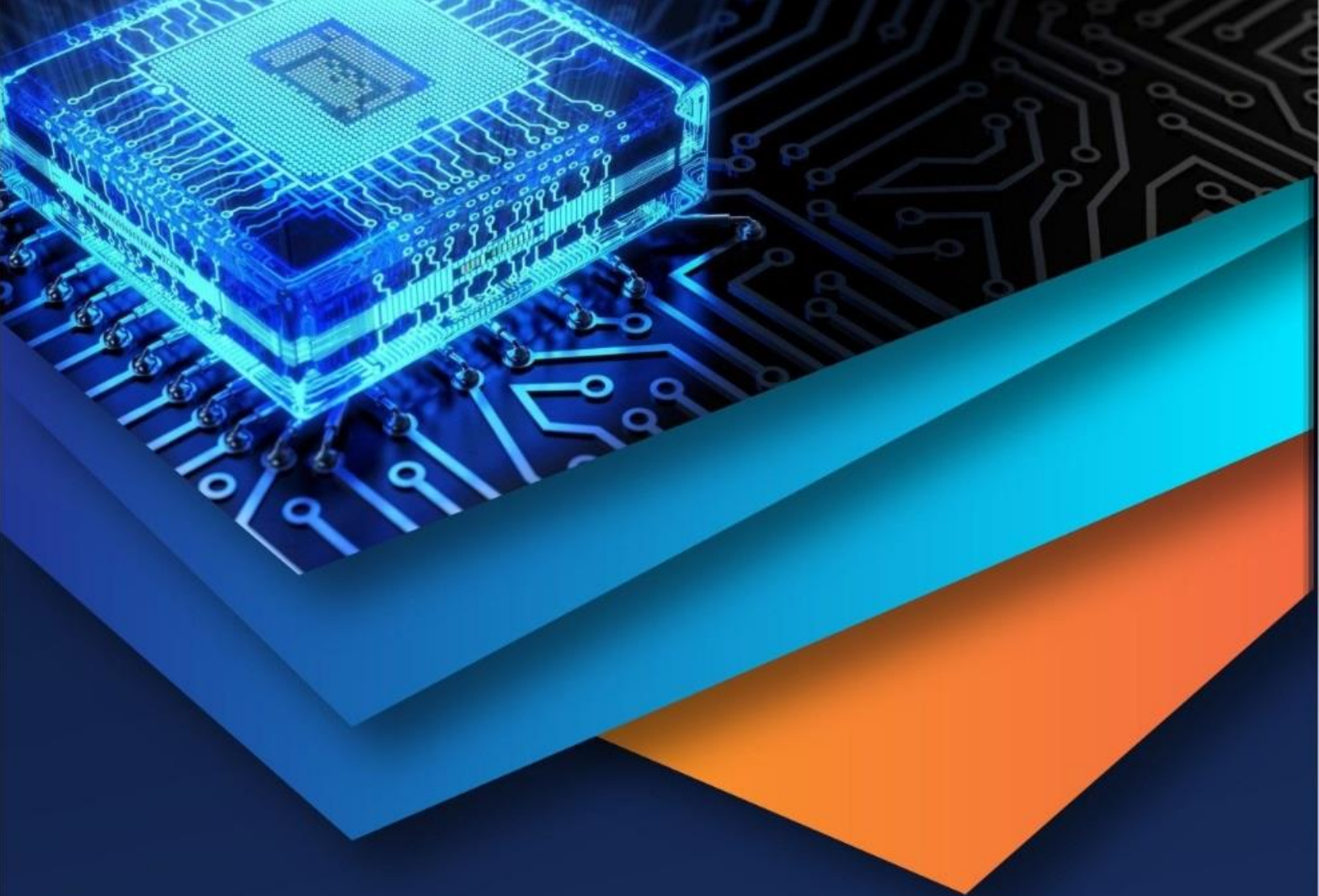
VI. CONCLUSION

This paper described Vakilo AI, a web-based platform that organizes the early stages of seeking legal help. Users are guided through case submission, the Gemini API supplies follow-up questions, a structured summary is prepared, and lawyers whose profiles have passed administrator review are recommended. Messaging, video consultation, appointment scheduling, document exchange, payment processing, and real-time notifications are all brought into the same workflow. The implementation draws on FastAPI, React, TypeScript, PostgreSQL, Supabase, Socket.IO, WebRTC, and Razorpay.

The project offers one practical way of linking AI-assisted intake with case coordination tools. Before stronger claims about real-world effectiveness can be made, however, the quality of the generated summaries and lawyer recommendations, along with the application's security, accessibility, and usability, must be evaluated systematically. Vakilo AI is meant to help users organize information and reach legal professionals, and it does not replace the judgment or advice of a qualified lawyer.

REFERENCES

- [1] Dr. Sudhakar Avareddy, Appu, Karthik M K, Mallikarjun, and Sai Shirisha, "Legal Assist: Smart Legal Case Management System," International Journal of Engineering Research & Technology (IJERT), Vol. 14 | Issue 12 | December 2025, Paper ID: IJERTV14IS120677.
- [2] Deepanshu, Divi Saxena, Deepali Rana, Ayesha Varshney, and Sahinur Rahman Laskar, "NyayaAI: An AI-Powered Legal Assistant Using Multi-Agent Architecture and Retrieval-Augmented Generation," arXiv preprint, Published: May 2026, arXiv:2605.10155, DOI: 10.48550/arXiv.2605.10155.
- [3] Amy J. Schmitz, Lola Akin Ojelabi, and John Zeleznikow, "Researching Online Dispute Resolution to Expand Access to Justice," GiustiziaConsensuale (Consensual Justice), Vol. 1, No. 2, 2021, pp. 269–301, Ohio State Legal Studies Research Paper No. 680, DOI: 10.2139/ssrn.4018593.
- [4] Shubham Kumar Nigam, Shubham Kumar Mishra, Ayush Kumar Mishra, Noel Shallum, and Arnab Bhattacharya, "Legal Question-Answering in the Indian Context: Efficacy, Challenges, and Potential of Modern AI Models," arXiv preprint, September 2023, arXiv:2309.14735.
- [5] T. Nava Krishna, B. Kusuma, K. Revanth, V. Reethi, K. Hemanth, and A. Prakash, "AI-Based Legal Information Retrieval Chatbot Using Natural Language Processing," International Journal of Engineering Research & Technology (IJERT), Vol. 15 | Issue 3 | March 2026, Paper ID: IJERTV15IS030786.
- [6] Jhanvi Arora, Tanay Patankar, Alay Shah, and Shubham Joshi, "Artificial Intelligence as Legal Research Assistant," FIRE Working Notes 2020, CEUR Workshop Proceedings, Vol. 2826, December 2020, pp. 60–65.
- [7] Yi Feng, Chuanyi Li, and Vincent Ng, "Legal Case Retrieval: A Survey of the State of the Art," Proceedings of the 62nd Annual Meeting of the Association for Computational Linguistics, 2024, pp. 6472–6485.
- [8] Jorge Martinez-Gil, "A Survey on Legal Question-Answering Systems," Computer Science Review, Vol. 48, May 2023, Article 100552, DOI: 10.1016/j.cosrev.2023.100552.
- [9] Rochak Bansal and Dr. Monica Chawla, "Transforming Legal Aid in India: Can AI Bridge the Justice Divide?" ShodhAI: Journal of Artificial Intelligence, Vol. 2 | Issue 1 | 2025, pp. 27–39, DOI: 10.29121/shodhai.v2.i1.2025.33.
- [10] Priyanka Prabhakar, Thanmai Gaddam, Divya Chennupalle, and Peeta Basa Pati, "Transformer-Based Embeddings and Stacking Classifiers for Judgment Outcome Prediction in Multilingual Legal Texts," Scientific Reports, Volume 16, Article Number 18305, Published: 20 April 2026, DOI: 10.1038/s41598-026-47652-x.
- [11] Raihan Khan, Sayak Sarkar, Sainik Kumar Mahata, and Edwin Jose, "Security Threats in Agentic AI System," arXiv preprint, October 2024, arXiv:2410.14728, DOI: 10.48550/arXiv.2410.14728.
- [12] Amy J. Schmitz, Lola Akin Ojelabi, and John Zeleznikow, "Researching Online Dispute Resolution to Expand Access to Justice," GiustiziaConsensuale (Consensual Justice), Vol. 1, No. 2, 2021, pp. 269–301, Ohio State Legal Studies Research Paper No. 680, DOI: 10.2139/ssrn.4018593.



10.22214/IJRASET



45.98



IMPACT FACTOR:
7.129



IMPACT FACTOR:
7.429



INTERNATIONAL JOURNAL FOR RESEARCH

IN APPLIED SCIENCE & ENGINEERING TECHNOLOGY

Call : 08813907089  (24*7 Support on Whatsapp)