PROJECT REPORT

LawStreet: AI-Powered Legal Assistant for Indian Law

Submitted by: **Team Null**

Group Members:

Nabarup Roy	: 241001011022
Priyanshu Kumar	: 231001001456
Ankur Kumar Bharti	: 241001011023
Ahad Alam	: 231001001452
Sreyasi Majumdar	: 231001001351
Muskan Kumari	: 23100100356
Sayan Biswas	: 231001001418

Department: Computer Science and Engineering
College: Techno India University

Academic Year: 2025-2026

Under the Guidance of:
Ratnadeep Dey
CSE

ABSTRACT

LawStreet is an AI-powered legal assistant platform designed to democratize access to Indian legal information. The system addresses the challenge of legal complexity and inaccessibility faced by common citizens who struggle to understand legal provisions without professional assistance.

The platform implements a **Retrieval-Augmented Generation (RAG)** pipeline that combines semantic search with Large Language Model (LLM) capabilities. The system ingests structured legal documents including the Indian Penal Code (IPC), Code of Criminal Procedure (CrPC), Constitution of India, Indian Evidence Act, and other statutes, converts them into vector embeddings using Sentence Transformers, and stores them in ChromaDB for efficient similarity search.

When a user submits a legal query through the React-based chat interface, the system retrieves the most relevant legal sections using semantic similarity matching and generates contextually accurate responses using LLaMA 3.3 via Ollama. Real-time communication is facilitated through Stream Chat SDK, enabling an interactive conversational experience.

Key Technologies: React 18, TypeScript, Node.js, Express, Python, FastAPI, ChromaDB, LLaMA 3.3, Sentence Transformers, Stream Chat

Outcome: A functional full-stack application capable of providing accurate, source-cited answers to legal queries, achieving response times of 2-5 seconds with 5 relevant document retrievals per query.

Keywords: RAG, Legal AI, NLP, Vector Database, LLM, Indian Law

Project Overview

Legal literacy remains a significant challenge in India due to the complexity of the legal system and the low population-to-lawyer ratio. Legal statutes such as the Indian Penal Code (IPC), Code of Criminal Procedure (CrPC), and the Constitution of India are written in highly technical language, making it difficult for common citizens—especially in rural areas—to understand their rights and legal obligations without professional assistance.

Recent advancements in Artificial Intelligence (AI) and Natural Language Processing (NLP) have created opportunities to improve access to legal information. Large Language Models (LLMs) such as GPT and LLaMA are capable of understanding and generating natural language responses. However, these models often suffer from hallucinations, where they generate responses that appear correct but lack factual accuracy—an issue that is critical in legal applications.

To address this problem, this project uses a Retrieval-Augmented Generation (RAG) approach. The system retrieves relevant legal sections from authenticated legal documents and uses them as context for generating responses, ensuring accuracy and source verification.

The primary objective of this project is to develop an AI-powered legal assistant that provides accurate, context-aware, and source-cited answers to queries related to Indian law. The system offers a modern web-based chat interface built using React and supports real-time communication through Stream Chat. A scalable backend architecture using Node.js manages requests, while a vector database powered by ChromaDB enables efficient semantic search. The system integrates LLaMA 3.3 via Ollama to perform local, privacy-friendly language model inference.

The scope of the project includes question answering for major Indian legal documents such as the IPC, CrPC, Constitution of India, and Evidence Act, along with features like dark/light mode support and citation-based responses. However, the system does not provide legal advice, does not analyze case laws or court judgments, and does not replace professional legal consultation.

LITERATURE REVIEW

10.1 Existing Systems

System	Description	Technology	Limitations
Indian Kanoon	Legal search engine	Keyword-based search	No natural language understanding, returns raw documents
ChatGPT / GPT-4	General-purpose LLM	Transformer-based	Hallucinations, no Indian law specialization, no citations
Ross Intelligence	AI legal research (US)	NLP + Legal DB	Discontinued, focused on US law only
Harvey AI	Legal AI for law firms	GPT-4 based	Commercial, expensive, not for public use
Casetext CoCounsel	Legal AI assistant	GPT-4 + Legal DB	US-focused, commercial, requires subscription

10.2 Limitations of Existing Systems

1. Keyword-based Search: Systems like Indian Kanoon rely on keyword matching, which fails to understand query intent and context.

2. Hallucination in LLMs: General-purpose LLMs like ChatGPT may generate plausible but incorrect legal information, which is dangerous in legal contexts.

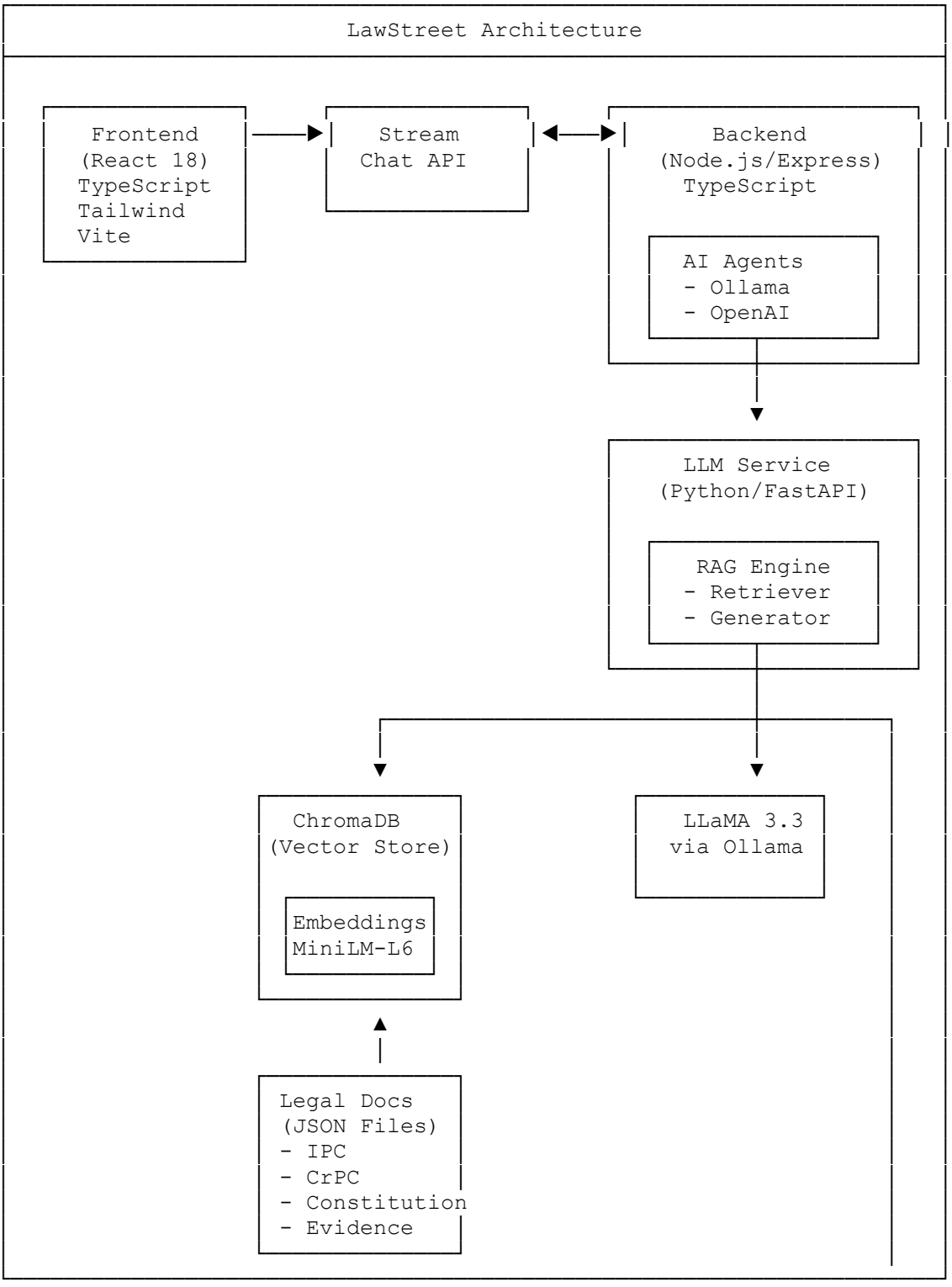
3. Lack of Indian Law Focus: Most advanced legal AI systems are designed for US or UK law and do not understand Indian legal provisions.

4. No Source Citation: Many AI systems do not provide citations, making it impossible to verify the accuracy of responses.

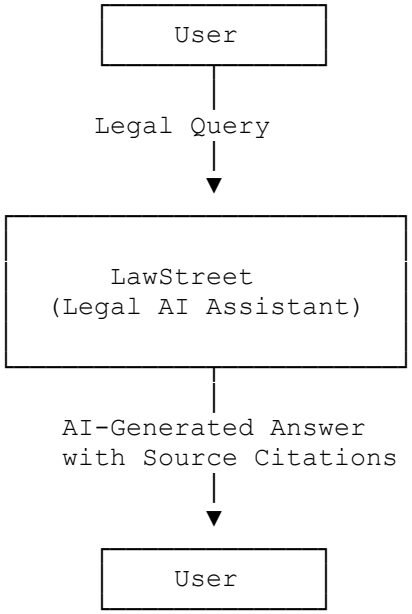
5. Commercial Barriers: Advanced legal AI tools are expensive and targeted at law firms, not individual citizens.

SYSTEM DESIGN

12.1 System Architecture



12.2 Data Flow Diagram (DFD)



TECHNOLOGY STACK

13.1 Frontend

Technology	Version	Purpose
React	18.3.1	UI Framework
TypeScript	5.5.3	Type-safe JavaScript
Vite	5.4.1	Build Tool
Tailwind CSS	3.4.11	Utility-first CSS
Stream Chat React	12.7.0	Real-time Chat
Radix UI	1.x	Accessible Components
Framer Motion	12.19.2	Animations
React Router	6.24.1	Client-side Routing

13.2 Backend

Technology	Version	Purpose
Node. js	20+	Runtime
Express	4.19.2	Web Framework
TypeScript	5.7.2	Type-safe JavaScript
Stream Chat SDK	8.46.0	Chat Backend
Axios	1.7.2	HTTP Client

13.3 LLM Service

Technology	Version	Purpose
Python	3.8+	Runtime
FastAPI	Latest	REST API Framework
ChromaDB	Latest	Vector Database
Sentence Transformers	Latest	Embedding Model
Ollama	Latest	Local LLM Inference
LLaMA 3.3	3.3	Large Language Model

13.4 Architecture Justification

Choice	Reason
RAG over Fine-tuning	Lower cost, no training data required, easy to update knowledge base
ChromaDB over Pinecone	Open-source, local deployment, no API costs
Ollama over OpenAI API	Privacy, no rate limits, no per-token costs
Stream Chat over Socket.io	Production-ready, scalable, excellent React SDK
Sentence Transformers	Open-source, high-quality embeddings, fast inference

IMPLEMENTATION

14.1 Module 1: Frontend (React Application)

The frontend is a single-page application built with React 18 and TypeScript.

14.2 Module 2: Backend (Node.js Server)

The backend manages AI agent lifecycle and provides authentication tokens.

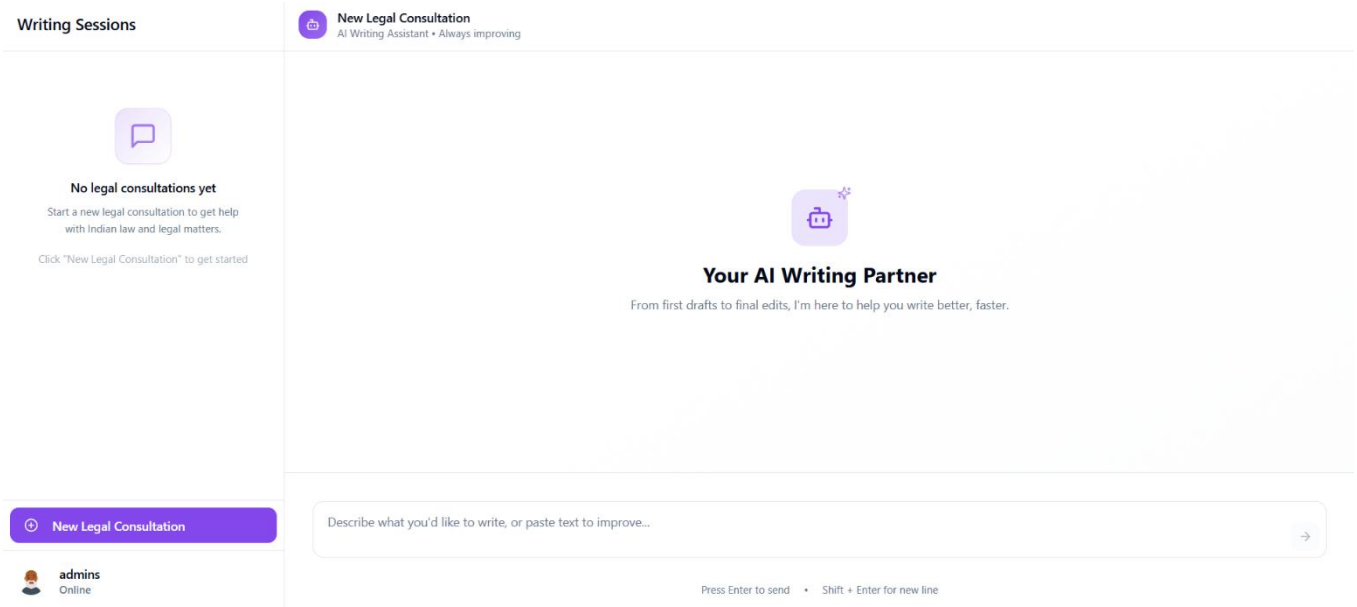
14.3 Module 3: LLM Service (Python RAG Pipeline)

The LLM service implements the core RAG pipeline.

RAG Pipeline Steps:
1. **Receive Query** - Accept user question via `/ask` endpoint
2. **Embed Query** - Convert to 384-dimensional vector using MiniLM-L6-v2
3. **Retrieve Documents** - Find top-5 similar documents in ChromaDB
4. **Build Prompt** - Construct prompt with context and question
5. **Generate Response** - Send to LLaMA 3.3 via Ollama
6. **Return Answer** - Include response and source citations

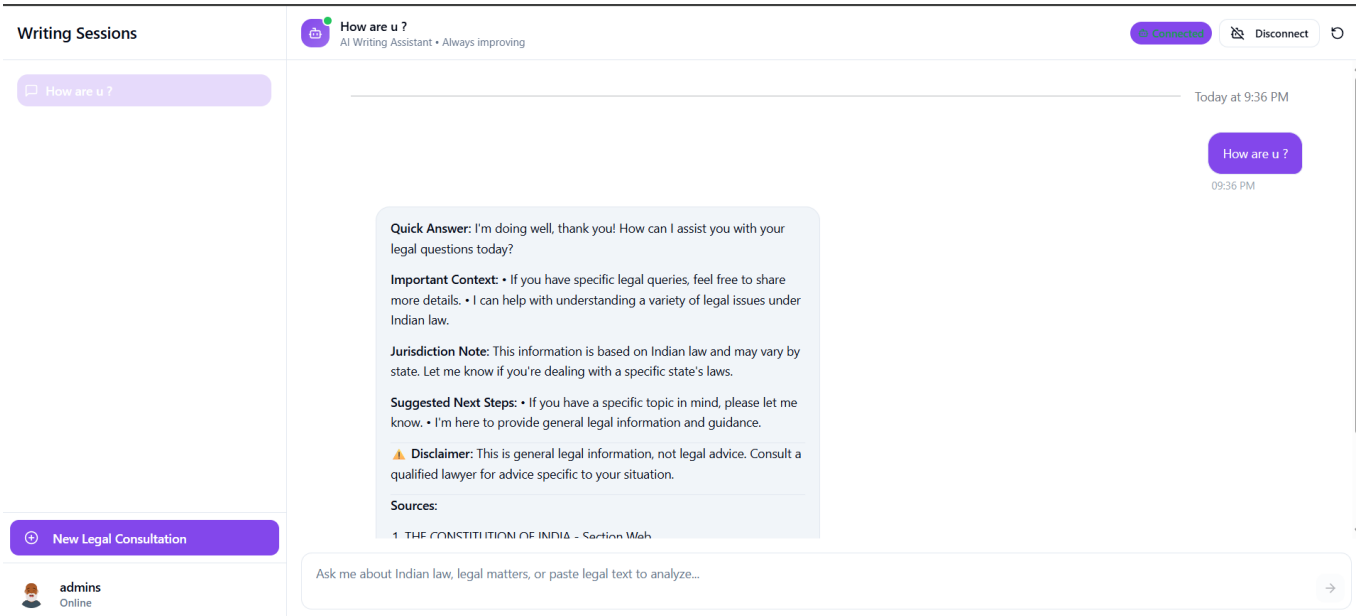
RESULTS & OUTPUT

15.1 Home Page



Description: The landing page displays the LawStreet logo, a brief description, and navigation to start a new chat session.

15.2 Chat Interface



Description: The chat interface shows the user's query on the right (user bubble) and the AI's response on the left (AI bubble). The interface supports markdown rendering for formatted responses.

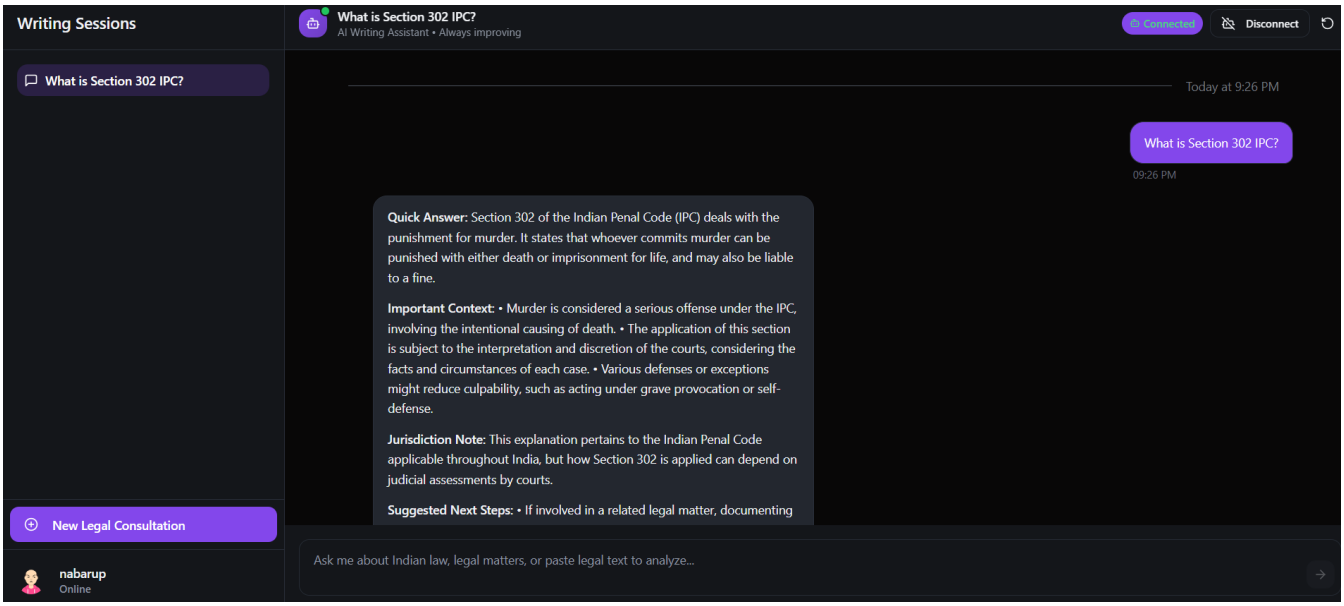
15.3 Sample Query and Response

Query: "What is Section 302 IPC?"

Response:
> Section 302 of the Indian Penal Code (IPC) deals with the punishment for murder. It states that whoever commits murder can be punished with either death or imprisonment for life, and may also be liable to a fine.
Important Context: • Murder is considered a serious offense under the IPC, involving the intentional causing of death. • The application of this section is subject to the interpretation and discretion of the courts, considering the facts and circumstances of

each case. • Various defenses or exceptions might reduce culpability, such as acting under grave provocation or self-defense.
Jurisdiction Note: This explanation pertains to the Indian Penal Code applicable throughout India, but how Section 302 is applied can depend on judicial assessments by courts.
Suggested Next Steps: • If involved in a related legal matter, documenting facts and circumstances is crucial. • Engage with legal counsel to ensure your rights and obligations are adequately represented. • Consider the specific legal implications and defenses by consulting a lawyer for comprehensive guidance.
Sources: IPC Section 302, IPC Section 300

15.4 Dark Mode Interface



Description: The application supports dark/light theme toggle for user preference and reduced eye strain.

CONCLUSION & FUTURE SCOPE

17.1 Conclusion

LawStreet successfully demonstrates the application of **Retrieval-Augmented Generation (RAG)** for creating an accessible legal information system for Indian law. The project achieves its objectives:

- Developed a functional RAG pipeline that retrieves relevant legal sections and generates accurate responses
- Built a modern, responsive React-based chat interface
- Implemented real-time communication using Stream Chat
- Created a scalable three-tier architecture (Frontend, Backend, LLM Service)
- Integrated LLaMA 3.3 for local, privacy-friendly inference
- Provided source citations for transparency and verification

The system demonstrates that AI can make legal information more accessible while maintaining accuracy through grounded generation.

17.2 Limitations

1. **No Legal Advice:** System provides information only, not legal counsel.
2. **Static Knowledge Base:** Does not automatically update with new amendments.
3. **English Only:** Currently supports only English queries.
4. **No Case Law:** Does not include court judgments or case analysis.
5. **Hardware Requirements:** Requires significant RAM for local LLM inference.

17.3 Future Scope

Enhancement	Description	Priority
Multi-language Support	Hindi and regional language queries	High
Voice Interface	Speech-to-text and text-to-speech	Medium
User Accounts	Save chat history and preferences	Medium
Document Upload	Allow users to upload contracts for analysis	Low

18. REFERENCES

[1] Lewis, P., Perez, E., Piktus, A., et al. (2020). "Retrieval-Augmented Generation for Knowledge-Intensive NLP Tasks."

[2] Touvron, H., et al. (2023). "LLaMA: Open and Efficient Foundation Language Models."

[3] Reimers, N., & Gurevych, I. (2019). "Sentence-BERT: Sentence Embeddings using Siamese BERT-Networks."

[4] ChromaDB Documentation. (2024). Retrieved from https://docs.trychroma.com

[5] FastAPI Documentation. (2024). Retrieved from https://fastapi.tiangolo.com

[6] Stream Chat Documentation. (2024). Retrieved from https://getstream.io/chat/docs/

[7] Indian Penal Code, 1860. Retrieved from https://indiacode. nic.in

[8] Code of Criminal Procedure, 1973. Retrieved from https://indiacode.nic.in

[9] Constitution of India. Retrieved from https://legislative.gov.in/constitution-of-india

[10] Ollama Documentation. (2024). Retrieved from https://ollama.ai

Thanks