

Key Points

- **Feasibility for a Student Project:** This idea is practical with open-source tools and Gemini's free API tier (limited to 5-15 requests per minute or up to 100 daily for advanced models), allowing basic development without billing. However, for heavier testing, consider local open-source AI alternatives like Llama models via Hugging Face to avoid rate limits.
- **Core Strengths:** The app promotes accessible, AI-driven mediation for small disputes, drawing on Indian Constitution resources like free public datasets, but emphasize it's not a legal substitute—include disclaimers to avoid liability.
- **Potential Challenges:** Handling sensitive data (proofs, statements) requires strong privacy measures; media analysis may strain free AI tiers, and electronic signatures need reliable open-source integration. Evidence suggests AI can hallucinate in legal contexts (up to 1 in 6 queries), so human oversight is recommended.
- **Open-Source Focus:** Prioritize free tools like React for UI, Node.js for backend, MongoDB for storage, and DocuSeal for signatures to keep costs at zero.

Project Overview

The AI Dispute Resolver is a web-based platform designed as a layered AI agent for resolving small civil disputes amicably. It allows users to file cases, submit evidence (including media), and receive AI-generated fair opinions based on Indian constitutional principles. If unresolved, cases escalate to formal channels. Built with open-source technologies, it uses Gemini AI for analysis, stores resolved cases for reference, and generates legal-style documents. This setup mimics mediation services but is non-binding, focusing on compromise.

Workflow

1. **Case Filing:** Party 1 registers, fills a form with dispute details, statements, and uploads

proofs (images, audio, video).

2. **Invitation:** System emails Party 2 (or multiple parties) to join and submit their side.
3. **Statement Collection:** Both parties provide inputs; AI reviews iteratively, prompting for additions until finalized.
4. **AI Analysis:** Gemini evaluates statements and media against Indian laws, generating top 3 compromise options.
5. **Resolution:**
 - Agreement on one option: Generate document, collect e-signatures, close case, and email copies.
 - Different options: Reanalyze and propose a refined best option.
 - No agreement: Forward details to a regional court (via email or API simulation).
6. **Storage:** Closed cases archived in database for future reference.

System Design

- **Architecture:** Client-server model with frontend for user interaction, backend for logic/AI

integration, and database for persistence.

- **Tech Stack** (All open-source/free):

Component	Tool/Library	Reason	🔗
Frontend	React.js	Interactive forms, file uploads; free and beginner-friendly.	
Backend	Node.js with Express	Handles API calls, email sending; scalable for students.	
Database	MongoDB (local or free Atlas tier)	Flexible for storing cases, statements, and media metadata.	
AI Integration	Google Gemini API (free tier)	For dispute analysis; embed Indian Constitution text from public sources like Hugging Face datasets.	
Email Service	Nodemailer with Gmail SMTP	Free email invites/notifications using app passwords.	
File Storage	Local server storage or Multer (Node.js)	Upload/handle media proofs without cloud billing.	
Electronic Signatures	DocuSeal or OpenSign	Open-source for e-signing generated PDFs.	
Document Generation	PDF-lib or Docassemble	Create court-like decision docs from templates.	

- **Data Flow:** User inputs → Backend validation → AI prompt with laws → Response parsing → Document gen/sign → DB store.
- **Security:** Use HTTPS, JWT for auth, encrypt sensitive data; host locally or on free platforms like Vercel (limited).

List of Improvements

See the detailed survey below for expanded suggestions.

The AI Dispute Resolver project envisions a user-friendly web application that leverages AI to facilitate out-of-court settlements for minor civil disputes, such as property

to facilitate out-of-court settlements for minor civil disputes, such as property disagreements, contractual issues, or neighbor conflicts, by analyzing submissions from involved parties and proposing equitable solutions grounded in Indian constitutional law. This concept draws inspiration from existing mediation platforms but incorporates AI for efficiency, making it accessible for non-legal users. Practically, as a student project, it can be prototyped in 4-6 weeks with basic coding skills, focusing on core features while iterating on edge cases like multi-party disputes or invalid proofs.

Brainstorming practical aspects reveals several considerations: First, user adoption hinges on trust, so include clear disclaimers stating the app provides opinions, not binding judgments, to mitigate legal risks—research indicates AI tools in law can err in 1/6 of queries due to hallucinations, emphasizing the need for source-based prompting. Second, media handling (images, audio, video) requires AI capabilities like Gemini's multimodal features (free tier supports this at limited scale), but for no-billing reliability, preprocess files locally with open-source libraries like FFmpeg for audio/video extraction before AI analysis. Third, integrating Indian laws: Use free resources such as the Hugging Face "Indian-Constitution" dataset or the India Code portal's digital repository of acts, which can be downloaded and embedded into AI prompts for context-aware responses. Avoid real-time API calls to paid legal databases; instead, cache static texts like the Constitution's full PDF (available via official government sites) and fine-tune prompts to reference Articles 14 (equality), 21 (life/liberty), or relevant civil codes.

The workflow can be refined for robustness: Start with user authentication via

email/password (no OAuth to avoid dependencies), then a guided form wizard for case filing—think step-by-step like Google Forms but with validation (e.g., required fields for dispute type, dates, involved parties). Email invitations should include secure links to join the case "room" (a unique ID in DB), preventing unauthorized access. During statement collection, implement a chat-like interface where AI (Gemini) intervenes dynamically: After initial submissions, it could query "Is this complete?" or "Provide more details on X," looping until both parties confirm finalization. For analysis, craft prompts like: "Based on Indian Constitution Articles Y-Z and provided statements/proofs, suggest top 3 compromise options for this dispute." Handle media by uploading to backend, then passing URLs(descriptions to Gemini for interpretation (e.g., "Analyze this image for evidence of property damage").

Resolution paths add practicality: If parties agree, use PDF-lib to template a "Settlement Agreement" document mimicking court formats (e.g., headers, clauses, party details), then integrate DocuSeal for e-signatures—users draw/sign via canvas, with timestamps for validity. Email distribution via Nodemailer ensures copies reach parties and a placeholder "lawyer" email. For disagreements, reanalysis could weight chosen options (e.g., "Party A picked Option 1, Party B Option 2—propose hybrid based on laws"). Escalation to courts: Simulate by generating a zipped case file (statements, proofs, AI opinions) and emailing to a mock regional court address, or integrate with free government portals if available. Database storage uses MongoDB schemas like {caseID, parties, statements, proofsURLs, AI_options, status, signatures}, enabling future queries for patterns (e.g., common dispute types) without advanced analytics.

System design emphasizes modularity for student development: A monolithic app initially, with potential microservices later. Frontend with React handles UI states (e.g., progress bars for AI processing), while backend routes manage CRUD operations. For AI, wrap Gemini calls in try-catch to handle rate limits—fallback to a local Ollama setup with a legal-fine-tuned model like LegalBERT if quotas exhaust. Performance: Limit concurrent cases to 5-10 for free tiers; use queues for AI requests. Testing: Mock emails with Mailtrap (free tier), unit test with Jest, and simulate disputes with sample data from public legal case studies. Deployment: Local via Docker for demo, or free hosts like Render.com's starter plan (no card needed initially).

Expanding on improvements, the list below incorporates practical enhancements based on similar open-source legal tools:

SIMILAR OPEN-SOURCE LEGAL TOOLS.

- **AI Alternatives for Scalability:** If Gemini's free tier (5-15 RPM, up to 100/day for Pro) proves limiting, integrate open-source models like Mistral-7B or Llama-3 via Hugging Face's free inference API or local Ollama—fine-tune on Indian law datasets for better accuracy without costs.
- **Enhanced Media Support:** Add validation for file types/sizes (e.g., <10MB) and use open-source tools like Tesseract OCR for image text extraction or SpeechRecognition for audio transcripts, feeding cleaned data to AI.
- **Legal Accuracy Boost:** Embed more resources, like Indian Kanoon's free API for case precedents, or download Bare Acts from India Code for comprehensive prompting; include a "confidence score" in AI outputs to flag uncertain analyses.
- **User Experience Upgrades:** Mobile-responsive design with React Native if expanding to apps; add real-time notifications via WebSockets (Socket.io, free) for party updates.
- **Privacy and Compliance:** Implement GDPR-like features (data deletion on request) using encryption libraries like Crypto.js; anonymize stored cases for future reference.
- **Multi-Language Support:** Since disputes may involve regional languages, use Gemini's translation or free libraries like Google Translate API (limited free) for Hindi/others.
- **Integration Expansions:** Link to free calendars (e.g., via iCal) for follow-up reminders; simulate court forwarding with email templates to actual Indian e-courts portals.
- **Analytics and Learning:** Add a dashboard for admins to view resolved case stats (e.g., success rate by dispute type) using free Chart.js; store anonymized data for AI fine-tuning.
- **Accessibility Features:** Ensure WCAG compliance with ARIA labels in React; support voice inputs via Web Speech API for differently-abled users.
- **Sustainability:** Open-source the project on GitHub for community contributions; include logging for debugging without external services.

This comprehensive approach transforms the idea into a viable, educational project, balancing innovation with practicality while adhering to open-source constraints.

Key Citations:

- Rate limits | Gemini API | Google AI for Developers - <https://ai.google.dev/gemini-api/docs/rate-limits>
- Gemini API Free Tier 2025: Complete Guide to Rate Limits & Models - <https://blog.laozhang.ai/api-guides/gemini-api-free-tier/>
- How to Use Google Gemini API Key for Free in 2025 - Apidog - <https://apidog.com/blog/google-gemini-api-key-for-free/>
- Evaluating AI for Law: Bridging the Gap with Open-Source Solutions - <https://arxiv.org/html/2404.12349v1>
- AI on Trial: Legal Models Hallucinate in 1 out of 6 (or More ...) - <https://hai.stanford.edu/news/ai-trial-legal-models-hallucinate-1-out-6-or-more-benchmarking-queries>
- India Legal Research Guide: Web Resources - NUS Libguides - <https://libguides.nus.edu.sg/c.php?g=145678&p=954730>
- India : legal resources: Legislation - Oxford LibGuides - <https://libguides.bodleian.ox.ac.uk/law-india/legislation>
- Indian Kanoon API - Home - <https://api.indiankanoon.org/>
- Sharathhebbar24/Indian-Constitution · Datasets at Hugging Face - <https://huggingface.co/datasets/Sharathhebbar24/Indian-Constitution>
- OpenSign™ | The Free & OpenSource Alternative to DocuSign - <https://www.opensignlabs.com/>
- DocuSeal | Open Source Document Signing - <https://www.docuseal.com/>
- Documenso - The Open Source DocuSign Alternative - <https://documenso.com/>
- Top 8 Woodpecker Alternatives for Legal Document Automation - <https://knackly.io/woodpecker-alternatives/>
- Best Legal Document Automation Software - Draftable - <https://www.draftable.com/draftable-legal-blog/the-best-legal-document-automation-software>
- Best Open Source PDF Generation Libraries 2025 - SourceForge - <https://sourceforge.net/directory/pdf-generation-libraries/>

<https://sourceforge.net/projects/pair-mediation/>

- ↳ Expand on multi-language support
- ↳ AI mediation in other countries