

# AI Legal Aid App - Implementation Roadmap

## Phase 0: Project Setup

- Create project root folder with frontend (Angular) and backend (Node.js + Express).
- Initialize Angular with routing, SCSS, and Node with Express and CORS.
- Verify setup with a simple test route.

## Phase 1: File Upload + Text Extraction

- Frontend: Build Angular form with file input (PDF/image), language dropdown, submit button.
- Backend: Use multer to handle file uploads to /uploads.
- For PDF: Extract text with pdf-parse. For images: Extract text with tesseract.js.
- Return extracted text to frontend for preview.

## Phase 2: AI Simplification

- Frontend: Add 'Simplify Text' button after preview.
- Backend: Integrate OpenAI GPT API with a prompt to simplify legal text.
- Return simplified text to frontend.

## Phase 3: Multi-Language Support

- Angular: Add ngx-translate for UI translations.
- Backend: Modify GPT prompt to return simplified text in Hindi (or other languages) if selected.

## Phase 4: History (Optional)

- Generate a userId and store in localStorage.
- Backend: Save document history (original text, simplified text, language, date) in MongoDB.
- Frontend: Create a history page to show past analyses.

## Phase 5: Polish & Extras

- Add speech-to-text for voice-based queries.
- Allow downloading simplified output as PDF.
- Add user authentication (Firebase Auth/Auth0).
- Make UI mobile-friendly with Angular Material or Bootstrap.

## Phase 6: Testing

- Test with real documents: rent agreements, FIR formats, traffic challans, legal notices.
- Verify PDF and image uploads, AI simplification in English and Hindi.
- Ensure OCR accuracy and clean UX.