# District Court Legal Application – Field-Level Postgres Schema, ERD & UI Mapping

This document provides a production-ready field-level PostgreSQL schema, an ERD-style relationship summary, and a mapping of fields to the current prototype screens (Dashboard, Cases, Deadlines, Rules Search, Judge Profiles, Alerts, Calendar, Settings).

## 1) PostgreSQL Schema (DDL)

Conventions: UUID primary keys, snake\_case, timestamptz, soft-delete via deleted\_at where appropriate. Enums shown as TEXT for portability (can be promoted to ENUM).

### users

CREATE TABLE users (  
 id UUID PRIMARY KEY DEFAULT gen\_random\_uuid(),  
 email TEXT UNIQUE NOT NULL,  
 full\_name TEXT NOT NULL,  
 role TEXT NOT NULL CHECK (role IN ('admin','lawyer','paralegal','guest')),  
 firm TEXT,  
 timezone TEXT DEFAULT 'America/Chicago',  
 mfa\_enabled BOOLEAN DEFAULT FALSE,  
 created\_at TIMESTAMPTZ NOT NULL DEFAULT now(),  
 updated\_at TIMESTAMPTZ NOT NULL DEFAULT now(),  
 deleted\_at TIMESTAMPTZ  
);

### courts

CREATE TABLE courts (  
 id UUID PRIMARY KEY DEFAULT gen\_random\_uuid(),  
 name TEXT NOT NULL,  
 district TEXT,  
 division TEXT,  
 location TEXT,  
 timezone TEXT NOT NULL,  
 website\_url TEXT  
);

### judges

CREATE TABLE judges (  
 id UUID PRIMARY KEY DEFAULT gen\_random\_uuid(),  
 full\_name TEXT NOT NULL,  
 court\_id UUID REFERENCES courts(id) ON DELETE SET NULL,  
 courtroom TEXT,  
 chambers\_url TEXT,  
 contact\_email TEXT,  
 contact\_phone TEXT  
);  
CREATE INDEX idx\_judges\_court ON judges(court\_id);

### judge\_procedures

CREATE TABLE judge\_procedures (  
 id UUID PRIMARY KEY DEFAULT gen\_random\_uuid(),  
 judge\_id UUID NOT NULL REFERENCES judges(id) ON DELETE CASCADE,  
 title TEXT NOT NULL,  
 version TEXT NOT NULL,  
 effective\_date DATE,  
 expiry\_date DATE,  
 content\_text TEXT,  
 filing\_format JSONB, -- e.g., single vs multi-PDF, naming  
 exhibit\_labeling JSONB,  
 motion\_practice JSONB,  
 filing\_cutoff\_time TIME,  
 hearing\_windows JSONB, -- e.g., available weekdays/times  
 created\_at TIMESTAMPTZ NOT NULL DEFAULT now()  
);  
CREATE INDEX idx\_proc\_judge ON judge\_procedures(judge\_id);

### cases

CREATE TABLE cases (  
 id UUID PRIMARY KEY DEFAULT gen\_random\_uuid(),  
 case\_number TEXT, -- official CM/ECF number  
 caption TEXT NOT NULL, -- case title  
 practice\_area TEXT,  
 court\_id UUID REFERENCES courts(id) ON DELETE SET NULL,  
 filing\_date DATE,  
 status TEXT CHECK (status IN ('open','stayed','closed','appeal','other')),  
 stage TEXT, -- pleadings, discovery, trial, etc.  
 lead\_attorney\_id UUID REFERENCES users(id) ON DELETE SET NULL,  
 confidentiality\_level TEXT,  
 legal\_hold BOOLEAN DEFAULT FALSE,  
 timezone TEXT NOT NULL,  
 created\_at TIMESTAMPTZ NOT NULL DEFAULT now(),  
 updated\_at TIMESTAMPTZ NOT NULL DEFAULT now(),  
 deleted\_at TIMESTAMPTZ  
);  
CREATE INDEX idx\_cases\_court ON cases(court\_id);  
CREATE INDEX idx\_cases\_lead ON cases(lead\_attorney\_id);

### case\_team

CREATE TABLE case\_team (  
 id UUID PRIMARY KEY DEFAULT gen\_random\_uuid(),  
 case\_id UUID NOT NULL REFERENCES cases(id) ON DELETE CASCADE,  
 user\_id UUID NOT NULL REFERENCES users(id) ON DELETE CASCADE,  
 role TEXT NOT NULL, -- e.g., owner, contributor, reviewer  
 added\_at TIMESTAMPTZ NOT NULL DEFAULT now(),  
 UNIQUE (case\_id, user\_id)  
);

### contacts

CREATE TABLE contacts (  
 id UUID PRIMARY KEY DEFAULT gen\_random\_uuid(),  
 type TEXT CHECK (type IN ('person','organization')) NOT NULL,  
 first\_name TEXT,  
 last\_name TEXT,  
 org\_name TEXT,  
 bar\_number TEXT,  
 email TEXT,  
 phone TEXT,  
 address JSONB,  
 preferred\_contact\_method TEXT,  
 availability JSONB  
);

### case\_contacts

CREATE TABLE case\_contacts (  
 id UUID PRIMARY KEY DEFAULT gen\_random\_uuid(),  
 case\_id UUID NOT NULL REFERENCES cases(id) ON DELETE CASCADE,  
 contact\_id UUID NOT NULL REFERENCES contacts(id) ON DELETE CASCADE,  
 role TEXT NOT NULL, -- opposing\_counsel, client, expert, vendor, reporter, other  
 UNIQUE (case\_id, contact\_id, role)  
);  
CREATE INDEX idx\_case\_contacts\_case ON case\_contacts(case\_id);

### documents

CREATE TABLE documents (  
 id UUID PRIMARY KEY DEFAULT gen\_random\_uuid(),  
 case\_id UUID REFERENCES cases(id) ON DELETE SET NULL,  
 source TEXT CHECK (source IN ('upload','pacer','web','judge\_procedure','rule')),  
 url TEXT,  
 file\_hash TEXT,  
 mime\_type TEXT,  
 page\_count INT,  
 ocr\_status TEXT CHECK (ocr\_status IN ('pending','complete','failed')) DEFAULT 'pending',  
 uploaded\_at TIMESTAMPTZ NOT NULL DEFAULT now()  
);

### doc\_chunks

CREATE TABLE doc\_chunks (  
 id UUID PRIMARY KEY DEFAULT gen\_random\_uuid(),  
 document\_id UUID NOT NULL REFERENCES documents(id) ON DELETE CASCADE,  
 chunk\_index INT NOT NULL,  
 start\_offset INT,  
 end\_offset INT,  
 heading TEXT,  
 embedding\_id UUID, -- optionally reference pgvector table or external  
 model\_version TEXT,  
 UNIQUE (document\_id, chunk\_index)  
);  
CREATE INDEX idx\_chunks\_doc ON doc\_chunks(document\_id);

### rules

CREATE TABLE rules (  
 id UUID PRIMARY KEY DEFAULT gen\_random\_uuid(),  
 source\_type TEXT CHECK (source\_type IN ('FRCP','LocalRule','JudgeProcedure','ECFManual','StandingOrder')) NOT NULL,  
 citation TEXT,  
 jurisdiction TEXT,  
 version TEXT,  
 effective\_date DATE,  
 superseded\_by UUID REFERENCES rules(id),  
 text TEXT,  
 url TEXT,  
 created\_at TIMESTAMPTZ NOT NULL DEFAULT now()  
);

### rule\_crossrefs

CREATE TABLE rule\_crossrefs (  
 from\_rule\_id UUID NOT NULL REFERENCES rules(id) ON DELETE CASCADE,  
 to\_rule\_id UUID NOT NULL REFERENCES rules(id) ON DELETE CASCADE,  
 PRIMARY KEY (from\_rule\_id, to\_rule\_id)  
);

### deadlines

CREATE TABLE deadlines (  
 id UUID PRIMARY KEY DEFAULT gen\_random\_uuid(),  
 case\_id UUID NOT NULL REFERENCES cases(id) ON DELETE CASCADE,  
 trigger\_type TEXT CHECK (trigger\_type IN ('rule','court\_order','user')) NOT NULL,  
 trigger\_source\_type TEXT, -- 'docket','rule','note'  
 trigger\_source\_id UUID,  
 basis TEXT CHECK (basis IN ('calendar\_days','business\_days')) NOT NULL,  
 holiday\_calendar\_id UUID,  
 due\_at TIMESTAMPTZ NOT NULL,  
 timezone TEXT NOT NULL,  
 owner\_id UUID REFERENCES users(id) ON DELETE SET NULL,  
 priority SMALLINT DEFAULT 3, -- 1=high 5=low  
 status TEXT CHECK (status IN ('open','snoozed','done','missed')) DEFAULT 'open',  
 snooze\_until TIMESTAMPTZ,  
 extension\_notes TEXT,  
 outcome TEXT, -- met / missed / extended summary  
 computation\_rationale TEXT,  
 created\_by UUID REFERENCES users(id),  
 updated\_by UUID REFERENCES users(id),  
 created\_at TIMESTAMPTZ NOT NULL DEFAULT now(),  
 updated\_at TIMESTAMPTZ NOT NULL DEFAULT now()  
);  
CREATE INDEX idx\_deadlines\_case\_due ON deadlines(case\_id, due\_at);

### deadline\_reminders

CREATE TABLE deadline\_reminders (  
 id UUID PRIMARY KEY DEFAULT gen\_random\_uuid(),  
 deadline\_id UUID NOT NULL REFERENCES deadlines(id) ON DELETE CASCADE,  
 notify\_at TIMESTAMPTZ NOT NULL,  
 channel TEXT CHECK (channel IN ('in\_app','email','sms','push')) NOT NULL,  
 sent BOOLEAN DEFAULT FALSE,  
 sent\_at TIMESTAMPTZ  
);  
CREATE INDEX idx\_reminders\_when ON deadline\_reminders(notify\_at, sent);

### deadline\_dependencies

CREATE TABLE deadline\_dependencies (  
 id UUID PRIMARY KEY DEFAULT gen\_random\_uuid(),  
 predecessor\_id UUID NOT NULL REFERENCES deadlines(id) ON DELETE CASCADE,  
 successor\_id UUID NOT NULL REFERENCES deadlines(id) ON DELETE CASCADE,  
 type TEXT, -- blocks, relates\_to  
 UNIQUE (predecessor\_id, successor\_id)  
);

### docket\_entries

CREATE TABLE docket\_entries (  
 id UUID PRIMARY KEY DEFAULT gen\_random\_uuid(),  
 case\_id UUID NOT NULL REFERENCES cases(id) ON DELETE CASCADE,  
 entry\_no INT,  
 entered\_at TIMESTAMPTZ,  
 entry\_type TEXT,  
 description TEXT,  
 ecf\_link TEXT,  
 pdf\_document\_id UUID REFERENCES documents(id)  
);  
CREATE INDEX idx\_docket\_case ON docket\_entries(case\_id);

### filings

CREATE TABLE filings (  
 id UUID PRIMARY KEY DEFAULT gen\_random\_uuid(),  
 case\_id UUID NOT NULL REFERENCES cases(id) ON DELETE CASCADE,  
 filing\_type TEXT,  
 ecf\_category TEXT,  
 packaged\_as TEXT CHECK (packaged\_as IN ('single\_pdf','multi\_pdf')),  
 served\_at TIMESTAMPTZ,  
 service\_method TEXT,  
 notes TEXT  
);

### filing\_exhibits

CREATE TABLE filing\_exhibits (  
 id UUID PRIMARY KEY DEFAULT gen\_random\_uuid(),  
 filing\_id UUID NOT NULL REFERENCES filings(id) ON DELETE CASCADE,  
 label TEXT NOT NULL, -- Exhibit A, B, etc.  
 document\_id UUID REFERENCES documents(id)  
);  
CREATE INDEX idx\_exhibits\_filing ON filing\_exhibits(filing\_id);

### service\_list

CREATE TABLE service\_list (  
 id UUID PRIMARY KEY DEFAULT gen\_random\_uuid(),  
 filing\_id UUID NOT NULL REFERENCES filings(id) ON DELETE CASCADE,  
 contact\_id UUID NOT NULL REFERENCES contacts(id) ON DELETE CASCADE  
);  
CREATE UNIQUE INDEX ux\_service\_pair ON service\_list(filing\_id, contact\_id);

### hearings

CREATE TABLE hearings (  
 id UUID PRIMARY KEY DEFAULT gen\_random\_uuid(),  
 case\_id UUID NOT NULL REFERENCES cases(id) ON DELETE CASCADE,  
 hearing\_type TEXT,  
 starts\_at TIMESTAMPTZ NOT NULL,  
 ends\_at TIMESTAMPTZ,  
 location TEXT,  
 virtual\_link TEXT,  
 requirements TEXT,  
 outcome TEXT  
);  
CREATE INDEX idx\_hearings\_case\_start ON hearings(case\_id, starts\_at);

### alerts

CREATE TABLE alerts (  
 id UUID PRIMARY KEY DEFAULT gen\_random\_uuid(),  
 case\_id UUID REFERENCES cases(id) ON DELETE SET NULL,  
 user\_id UUID NOT NULL REFERENCES users(id) ON DELETE CASCADE,  
 type TEXT,  
 severity TEXT, -- info/warn/critical  
 title TEXT,  
 body TEXT,  
 created\_at TIMESTAMPTZ NOT NULL DEFAULT now(),  
 read\_at TIMESTAMPTZ  
);  
CREATE INDEX idx\_alerts\_user\_created ON alerts(user\_id, created\_at DESC);

### calendar\_events

CREATE TABLE calendar\_events (  
 id UUID PRIMARY KEY DEFAULT gen\_random\_uuid(),  
 case\_id UUID REFERENCES cases(id) ON DELETE SET NULL,  
 external\_provider TEXT CHECK (external\_provider IN ('google','m365','ical')),  
 external\_event\_id TEXT,  
 starts\_at TIMESTAMPTZ NOT NULL,  
 ends\_at TIMESTAMPTZ,  
 sync\_state TEXT,  
 last\_synced\_at TIMESTAMPTZ  
);  
CREATE INDEX idx\_cal\_events\_case\_start ON calendar\_events(case\_id, starts\_at);

### holiday\_calendars

CREATE TABLE holiday\_calendars (  
 id UUID PRIMARY KEY DEFAULT gen\_random\_uuid(),  
 name TEXT NOT NULL,  
 jurisdiction TEXT,  
 timezone TEXT NOT NULL,  
 source\_url TEXT  
);

### holidays

CREATE TABLE holidays (  
 id UUID PRIMARY KEY DEFAULT gen\_random\_uuid(),  
 calendar\_id UUID NOT NULL REFERENCES holiday\_calendars(id) ON DELETE CASCADE,  
 date DATE NOT NULL,  
 name TEXT NOT NULL,  
 UNIQUE (calendar\_id, date)  
);

### user\_notification\_subscriptions

CREATE TABLE user\_notification\_subscriptions (  
 id UUID PRIMARY KEY DEFAULT gen\_random\_uuid(),  
 user\_id UUID NOT NULL REFERENCES users(id) ON DELETE CASCADE,  
 case\_id UUID REFERENCES cases(id) ON DELETE CASCADE,  
 type TEXT NOT NULL, -- deadlines, docket, hearings, all  
 UNIQUE (user\_id, COALESCE(case\_id, '00000000-0000-0000-0000-000000000000')::uuid, type)  
);

### notifications\_log

CREATE TABLE notifications\_log (  
 id UUID PRIMARY KEY DEFAULT gen\_random\_uuid(),  
 user\_id UUID NOT NULL REFERENCES users(id) ON DELETE CASCADE,  
 channel TEXT CHECK (channel IN ('in\_app','email','sms','push')) NOT NULL,  
 payload JSONB,  
 delivered\_at TIMESTAMPTZ,  
 status TEXT CHECK (status IN ('queued','sent','failed')) NOT NULL DEFAULT 'queued'  
);

### audit\_log

CREATE TABLE audit\_log (  
 id UUID PRIMARY KEY DEFAULT gen\_random\_uuid(),  
 actor\_user\_id UUID REFERENCES users(id),  
 entity\_table TEXT NOT NULL,  
 entity\_id UUID NOT NULL,  
 action TEXT NOT NULL, -- create/update/delete/compute  
 before JSONB,  
 after JSONB,  
 created\_at TIMESTAMPTZ NOT NULL DEFAULT now()  
);  
CREATE INDEX idx\_audit\_entity ON audit\_log(entity\_table, entity\_id);

## 2) ERD – Relationship Summary

- courts 1─\* judges; courts 1─\* cases

- judges 1─\* judge\_procedures

- cases 1─\* case\_team (to users); users \*─\* cases via case\_team

- cases 1─\* case\_contacts (to contacts) with role; contacts reused across cases

- cases 1─\* documents; documents 1─\* doc\_chunks

- rules \*─\* rules via rule\_crossrefs (self-referencing)

- cases 1─\* deadlines; deadlines 1─\* deadline\_reminders; deadlines \*─\* deadlines via deadline\_dependencies

- cases 1─\* docket\_entries; cases 1─\* filings; filings 1─\* filing\_exhibits; filings \*─\* contacts via service\_list

- cases 1─\* hearings; cases 1─\* calendar\_events

- users 1─\* alerts; users 1─\* notifications\_log; users 1─\* user\_notification\_subscriptions

- holiday\_calendars 1─\* holidays; deadlines ─> holiday\_calendars (optional)

- audit\_log references any entity by (entity\_table, entity\_id)

## 3) Field-to-UI Screen Mapping

### Dashboard

|  |  |  |
| --- | --- | --- |
| Screen / Component | Entity/Table | Key Fields |
| Dashboard | cases | status, stage, filing\_date |
| Dashboard | deadlines | due\_at, status, priority, case\_id |
| Dashboard | deadlines + alerts | priority=1, status!='done' ; severity='critical' |
| Dashboard | judges | full\_name, court\_id |

### Cases

|  |  |  |
| --- | --- | --- |
| Screen / Component | Entity/Table | Key Fields |
| Cases List | cases | caption, case\_number, status, court\_id, lead\_attorney\_id |
| Case Detail: Parties | case\_contacts + contacts | role, first\_name/last\_name/org\_name, bar\_number, email, phone |
| Case Detail: Team | case\_team + users | role, full\_name, email |
| Case Detail: Docket | docket\_entries | entry\_no, entered\_at, entry\_type, description |
| Case Detail: Filings | filings + filing\_exhibits | filing\_type, ecf\_category, packaged\_as, exhibits |
| Case Detail: Documents | documents | source, url, mime\_type, ocr\_status |

### Deadline Tracker

|  |  |  |
| --- | --- | --- |
| Screen / Component | Entity/Table | Key Fields |
| Deadlines Table | deadlines | trigger\_type, due\_at, owner\_id, status, priority |
| Reminders | deadline\_reminders | notify\_at, channel, sent |
| Dependencies | deadline\_dependencies | predecessor\_id, successor\_id, type |

### Rules & Research

|  |  |  |
| --- | --- | --- |
| Screen / Component | Entity/Table | Key Fields |
| Rules Search | rules | source\_type, citation, jurisdiction, version, text |
| Related Rules | rule\_crossrefs | from\_rule\_id, to\_rule\_id |
| Q&A Sources | doc\_chunks | document\_id, heading, model\_version |

### Judge Profiles

|  |  |  |
| --- | --- | --- |
| Screen / Component | Entity/Table | Key Fields |
| Judges List | judges | full\_name, court\_id, courtroom |
| Judge Profile | judge\_procedures | title, version, effective\_date, filing\_format, motion\_practice |

### Alerts

|  |  |  |
| --- | --- | --- |
| Screen / Component | Entity/Table | Key Fields |
| Alerts Center | alerts | type, severity, title, created\_at, read\_at |

### Calendar

|  |  |  |
| --- | --- | --- |
| Screen / Component | Entity/Table | Key Fields |
| Calendar | calendar\_events + hearings + deadlines | starts\_at/ends\_at; hearing\_type; due\_at (rendered as all-day) |

### Settings

|  |  |  |
| --- | --- | --- |
| Screen / Component | Entity/Table | Key Fields |
| User Settings | users | full\_name, email, timezone, role |
| Notifications Preferences | user\_notification\_subscriptions | type, case\_id |