-- =====================================================

-- LAW FIRM MANAGEMENT PLATFORM - SaaS DATABASE SCHEMA

-- Multi-Tenant Architecture with Super Admin Control

-- =====================================================

-- =====================================================

-- 1. PLATFORM MANAGEMENT (Super Admin Level)

-- =====================================================

-- Platform configuration and settings

CREATE TABLE platform\_settings (

id UUID PRIMARY KEY DEFAULT gen\_random\_uuid(),

key VARCHAR(100) UNIQUE NOT NULL,

value JSONB NOT NULL,

description TEXT,

created\_at TIMESTAMP WITH TIME ZONE DEFAULT NOW(),

updated\_at TIMESTAMP WITH TIME ZONE DEFAULT NOW()

);

-- Platform users (Super Admin, Support Staff)

CREATE TABLE platform\_users (

id UUID PRIMARY KEY DEFAULT gen\_random\_uuid(),

email VARCHAR(255) UNIQUE NOT NULL,

password\_hash VARCHAR(255) NOT NULL,

first\_name VARCHAR(100) NOT NULL,

last\_name VARCHAR(100) NOT NULL,

role VARCHAR(50) NOT NULL DEFAULT 'support', -- super\_admin, support, billing\_admin

permissions JSONB DEFAULT '[]', -- platform-level permissions

is\_active BOOLEAN DEFAULT true,

last\_login\_at TIMESTAMP WITH TIME ZONE,

created\_at TIMESTAMP WITH TIME ZONE DEFAULT NOW(),

updated\_at TIMESTAMP WITH TIME ZONE DEFAULT NOW()

);

-- Platform sessions for super admin authentication

CREATE TABLE platform\_sessions (

id UUID PRIMARY KEY DEFAULT gen\_random\_uuid(),

platform\_user\_id UUID NOT NULL REFERENCES platform\_users(id) ON DELETE CASCADE,

session\_token VARCHAR(255) UNIQUE NOT NULL,

ip\_address INET,

user\_agent TEXT,

expires\_at TIMESTAMP WITH TIME ZONE NOT NULL,

created\_at TIMESTAMP WITH TIME ZONE DEFAULT NOW()

);

-- Subscription plans available on the platform

CREATE TABLE subscription\_plans (

id UUID PRIMARY KEY DEFAULT gen\_random\_uuid(),

name VARCHAR(100) NOT NULL,

slug VARCHAR(100) UNIQUE NOT NULL,

description TEXT,

price\_monthly DECIMAL(10,2),

price\_yearly DECIMAL(10,2),

max\_users INTEGER,

max\_cases INTEGER,

max\_storage\_gb INTEGER,

features JSONB DEFAULT '[]', -- array of included features

is\_active BOOLEAN DEFAULT true,

created\_at TIMESTAMP WITH TIME ZONE DEFAULT NOW(),

updated\_at TIMESTAMP WITH TIME ZONE DEFAULT NOW()

);

-- =====================================================

-- 2. LAW FIRM MANAGEMENT (Organization Level)

-- =====================================================

-- Law firms (tenants) on the platform

CREATE TABLE law\_firms (

id UUID PRIMARY KEY DEFAULT gen\_random\_uuid(),

name VARCHAR(255) NOT NULL,

slug VARCHAR(100) UNIQUE NOT NULL, -- for subdomain/url routing

logo\_url VARCHAR(500),

address TEXT,

phone VARCHAR(20),

email VARCHAR(255),

website VARCHAR(255),

timezone VARCHAR(50) DEFAULT 'UTC',

-- Subscription details

subscription\_plan\_id UUID REFERENCES subscription\_plans(id),

subscription\_status VARCHAR(20) DEFAULT 'trial', -- trial, active, suspended, cancelled

subscription\_started\_at TIMESTAMP WITH TIME ZONE,

subscription\_ends\_at TIMESTAMP WITH TIME ZONE,

-- Billing information

billing\_email VARCHAR(255),

billing\_address TEXT,

stripe\_customer\_id VARCHAR(255),

-- Usage limits and tracking

max\_users INTEGER DEFAULT 10,

max\_cases INTEGER DEFAULT 100,

max\_storage\_gb INTEGER DEFAULT 5,

current\_user\_count INTEGER DEFAULT 0,

current\_case\_count INTEGER DEFAULT 0,

current\_storage\_gb DECIMAL(10,2) DEFAULT 0,

-- Platform management

onboarded\_by UUID REFERENCES platform\_users(id),

is\_active BOOLEAN DEFAULT true,

settings JSONB DEFAULT '{}', -- firm-specific settings

created\_at TIMESTAMP WITH TIME ZONE DEFAULT NOW(),

updated\_at TIMESTAMP WITH TIME ZONE DEFAULT NOW()

);

-- Law firm billing history

CREATE TABLE law\_firm\_billing (

id UUID PRIMARY KEY DEFAULT gen\_random\_uuid(),

law\_firm\_id UUID NOT NULL REFERENCES law\_firms(id) ON DELETE CASCADE,

subscription\_plan\_id UUID REFERENCES subscription\_plans(id),

billing\_period\_start DATE NOT NULL,

billing\_period\_end DATE NOT NULL,

amount DECIMAL(10,2) NOT NULL,

tax\_amount DECIMAL(10,2) DEFAULT 0,

total\_amount DECIMAL(10,2) NOT NULL,

currency VARCHAR(3) DEFAULT 'USD',

status VARCHAR(20) DEFAULT 'pending', -- pending, paid, failed, refunded

stripe\_invoice\_id VARCHAR(255),

paid\_at TIMESTAMP WITH TIME ZONE,

created\_at TIMESTAMP WITH TIME ZONE DEFAULT NOW()

);

-- Platform-level audit logs for super admin activities

CREATE TABLE platform\_audit\_logs (

id UUID PRIMARY KEY DEFAULT gen\_random\_uuid(),

platform\_user\_id UUID REFERENCES platform\_users(id),

law\_firm\_id UUID REFERENCES law\_firms(id), -- if action was on a specific firm

action VARCHAR(100) NOT NULL,

entity\_type VARCHAR(50) NOT NULL, -- law\_firm, subscription, user, etc.

entity\_id UUID,

old\_values JSONB,

new\_values JSONB,

ip\_address INET,

user\_agent TEXT,

created\_at TIMESTAMP WITH TIME ZONE DEFAULT NOW()

);

-- Workspaces within law firms (departments, teams, practice areas)

CREATE TABLE workspaces (

id UUID PRIMARY KEY DEFAULT gen\_random\_uuid(),

law\_firm\_id UUID NOT NULL REFERENCES law\_firms(id) ON DELETE CASCADE,

name VARCHAR(255) NOT NULL,

description TEXT,

workspace\_type VARCHAR(50) DEFAULT 'department', -- department, team, practice\_area

parent\_workspace\_id UUID REFERENCES workspaces(id), -- for hierarchical workspaces

settings JSONB DEFAULT '{}',

is\_active BOOLEAN DEFAULT true,

created\_at TIMESTAMP WITH TIME ZONE DEFAULT NOW(),

updated\_at TIMESTAMP WITH TIME ZONE DEFAULT NOW()

);

-- =====================================================

-- 3. LAW FIRM USER MANAGEMENT

-- =====================================================

-- Users within law firms (lawyers, staff, clients)

CREATE TABLE users (

id UUID PRIMARY KEY DEFAULT gen\_random\_uuid(),

law\_firm\_id UUID NOT NULL REFERENCES law\_firms(id) ON DELETE CASCADE,

email VARCHAR(255) NOT NULL, -- unique within firm context

password\_hash VARCHAR(255) NOT NULL,

first\_name VARCHAR(100) NOT NULL,

last\_name VARCHAR(100) NOT NULL,

phone VARCHAR(20),

avatar\_url VARCHAR(500),

employee\_id VARCHAR(50), -- internal employee ID

bar\_license\_number VARCHAR(100), -- for lawyers

specializations TEXT[], -- legal specializations

user\_type VARCHAR(20) DEFAULT 'internal', -- internal, client

is\_active BOOLEAN DEFAULT true,

email\_verified BOOLEAN DEFAULT false,

last\_login\_at TIMESTAMP WITH TIME ZONE,

password\_changed\_at TIMESTAMP WITH TIME ZONE DEFAULT NOW(),

created\_at TIMESTAMP WITH TIME ZONE DEFAULT NOW(),

updated\_at TIMESTAMP WITH TIME ZONE DEFAULT NOW(),

UNIQUE(law\_firm\_id, email) -- email unique within each law firm

);

-- Password reset tokens for law firm users

CREATE TABLE password\_reset\_tokens (

id UUID PRIMARY KEY DEFAULT gen\_random\_uuid(),

user\_id UUID NOT NULL REFERENCES users(id) ON DELETE CASCADE,

token VARCHAR(255) UNIQUE NOT NULL,

expires\_at TIMESTAMP WITH TIME ZONE NOT NULL,

used\_at TIMESTAMP WITH TIME ZONE,

created\_at TIMESTAMP WITH TIME ZONE DEFAULT NOW()

);

-- User sessions for law firm users

CREATE TABLE user\_sessions (

id UUID PRIMARY KEY DEFAULT gen\_random\_uuid(),

user\_id UUID NOT NULL REFERENCES users(id) ON DELETE CASCADE,

law\_firm\_id UUID NOT NULL REFERENCES law\_firms(id) ON DELETE CASCADE,

session\_token VARCHAR(255) UNIQUE NOT NULL,

refresh\_token VARCHAR(255) UNIQUE,

ip\_address INET,

user\_agent TEXT,

expires\_at TIMESTAMP WITH TIME ZONE NOT NULL,

last\_activity\_at TIMESTAMP WITH TIME ZONE DEFAULT NOW(),

created\_at TIMESTAMP WITH TIME ZONE DEFAULT NOW()

);

-- =====================================================

-- 4. ROLE-BASED ACCESS CONTROL (Multi-Level)

-- =====================================================

-- Define permissions available in the system

CREATE TABLE permissions (

id UUID PRIMARY KEY DEFAULT gen\_random\_uuid(),

name VARCHAR(100) UNIQUE NOT NULL,

slug VARCHAR(100) UNIQUE NOT NULL,

category VARCHAR(50) NOT NULL, -- cases, users, documents, calendar, etc.

description TEXT,

is\_platform\_permission BOOLEAN DEFAULT false, -- platform vs firm-level permission

created\_at TIMESTAMP WITH TIME ZONE DEFAULT NOW()

);

-- Define roles within law firms

CREATE TABLE roles (

id UUID PRIMARY KEY DEFAULT gen\_random\_uuid(),

law\_firm\_id UUID NOT NULL REFERENCES law\_firms(id) ON DELETE CASCADE,

name VARCHAR(100) NOT NULL,

slug VARCHAR(100) NOT NULL, -- owner, senior\_lawyer, junior\_lawyer, assistant, secretary, client

description TEXT,

level INTEGER NOT NULL DEFAULT 0, -- hierarchy level (higher = more permissions)

is\_system\_role BOOLEAN DEFAULT false, -- system-defined vs custom roles

permissions JSONB DEFAULT '[]', -- array of permission strings

created\_at TIMESTAMP WITH TIME ZONE DEFAULT NOW(),

updated\_at TIMESTAMP WITH TIME ZONE DEFAULT NOW(),

UNIQUE(law\_firm\_id, slug)

);

-- Many-to-many relationship between roles and permissions

CREATE TABLE role\_permissions (

id UUID PRIMARY KEY DEFAULT gen\_random\_uuid(),

role\_id UUID NOT NULL REFERENCES roles(id) ON DELETE CASCADE,

permission\_id UUID NOT NULL REFERENCES permissions(id) ON DELETE CASCADE,

created\_at TIMESTAMP WITH TIME ZONE DEFAULT NOW(),

UNIQUE(role\_id, permission\_id)

);

-- User role assignments within law firms

CREATE TABLE user\_roles (

id UUID PRIMARY KEY DEFAULT gen\_random\_uuid(),

user\_id UUID NOT NULL REFERENCES users(id) ON DELETE CASCADE,

role\_id UUID NOT NULL REFERENCES roles(id) ON DELETE CASCADE,

workspace\_id UUID REFERENCES workspaces(id), -- role within specific workspace

assigned\_by UUID REFERENCES users(id),

assigned\_at TIMESTAMP WITH TIME ZONE DEFAULT NOW(),

expires\_at TIMESTAMP WITH TIME ZONE, -- optional role expiration

is\_active BOOLEAN DEFAULT true,

UNIQUE(user\_id, role\_id, workspace\_id)

);

-- Workspace access control within law firms

CREATE TABLE workspace\_members (

id UUID PRIMARY KEY DEFAULT gen\_random\_uuid(),

workspace\_id UUID NOT NULL REFERENCES workspaces(id) ON DELETE CASCADE,

user\_id UUID NOT NULL REFERENCES users(id) ON DELETE CASCADE,

role\_in\_workspace VARCHAR(50) DEFAULT 'member', -- owner, admin, member, viewer

added\_by UUID REFERENCES users(id),

joined\_at TIMESTAMP WITH TIME ZONE DEFAULT NOW(),

is\_active BOOLEAN DEFAULT true,

UNIQUE(workspace\_id, user\_id)

);

-- =====================================================

-- 5. CASE MANAGEMENT (Law Firm Scoped)

-- =====================================================

-- Case categories/types within law firms

CREATE TABLE case\_categories (

id UUID PRIMARY KEY DEFAULT gen\_random\_uuid(),

law\_firm\_id UUID NOT NULL REFERENCES law\_firms(id) ON DELETE CASCADE,

name VARCHAR(100) NOT NULL,

description TEXT,

color VARCHAR(7), -- hex color for UI

is\_active BOOLEAN DEFAULT true,

created\_at TIMESTAMP WITH TIME ZONE DEFAULT NOW(),

UNIQUE(law\_firm\_id, name)

);

-- Main cases table (law firm scoped)

CREATE TABLE cases (

id UUID PRIMARY KEY DEFAULT gen\_random\_uuid(),

law\_firm\_id UUID NOT NULL REFERENCES law\_firms(id) ON DELETE CASCADE,

workspace\_id UUID REFERENCES workspaces(id),

case\_number VARCHAR(100) NOT NULL, -- unique within law firm

title VARCHAR(255) NOT NULL,

description TEXT,

case\_category\_id UUID REFERENCES case\_categories(id),

case\_type VARCHAR(50) NOT NULL, -- civil, criminal, corporate, family, etc.

status VARCHAR(50) DEFAULT 'active', -- active, pending, closed, archived

priority VARCHAR(20) DEFAULT 'medium', -- low, medium, high, urgent

estimated\_value DECIMAL(15,2),

actual\_value DECIMAL(15,2),

client\_id UUID REFERENCES users(id), -- primary client

lead\_lawyer\_id UUID REFERENCES users(id), -- primary lawyer

court\_name VARCHAR(255),

judge\_name VARCHAR(255),

opposing\_party VARCHAR(255),

opposing\_counsel VARCHAR(255),

statute\_of\_limitations DATE,

case\_opened\_date DATE,

case\_closed\_date DATE,

is\_billable BOOLEAN DEFAULT true,

billing\_rate DECIMAL(10,2),

metadata JSONB DEFAULT '{}', -- flexible field for case-specific data

created\_by UUID NOT NULL REFERENCES users(id),

created\_at TIMESTAMP WITH TIME ZONE DEFAULT NOW(),

updated\_at TIMESTAMP WITH TIME ZONE DEFAULT NOW(),

UNIQUE(law\_firm\_id, case\_number)

);

-- Case team assignments (multiple lawyers/staff per case)

CREATE TABLE case\_assignments (

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\_in\_case VARCHAR(50) NOT NULL, -- lead\_lawyer, associate, paralegal, assistant

assigned\_by UUID REFERENCES users(id),

assigned\_at TIMESTAMP WITH TIME ZONE DEFAULT NOW(),

removed\_at TIMESTAMP WITH TIME ZONE,

is\_active BOOLEAN DEFAULT true,

UNIQUE(case\_id, user\_id, role\_in\_case)

);

-- Case status history for tracking

CREATE TABLE case\_status\_history (

id UUID PRIMARY KEY DEFAULT gen\_random\_uuid(),

case\_id UUID NOT NULL REFERENCES cases(id) ON DELETE CASCADE,

old\_status VARCHAR(50),

new\_status VARCHAR(50) NOT NULL,

reason TEXT,

changed\_by UUID NOT NULL REFERENCES users(id),

changed\_at TIMESTAMP WITH TIME ZONE DEFAULT NOW()

);

-- =====================================================

-- 6. DOCUMENT MANAGEMENT (Law Firm Scoped)

-- =====================================================

-- Document categories within law firms

CREATE TABLE document\_categories (

id UUID PRIMARY KEY DEFAULT gen\_random\_uuid(),

law\_firm\_id UUID NOT NULL REFERENCES law\_firms(id) ON DELETE CASCADE,

name VARCHAR(100) NOT NULL,

description TEXT,

is\_active BOOLEAN DEFAULT true,

created\_at TIMESTAMP WITH TIME ZONE DEFAULT NOW(),

UNIQUE(law\_firm\_id, name)

);

-- Documents table (law firm scoped with tenant isolation)

CREATE TABLE documents (

id UUID PRIMARY KEY DEFAULT gen\_random\_uuid(),

law\_firm\_id UUID NOT NULL REFERENCES law\_firms(id) ON DELETE CASCADE,

case\_id UUID REFERENCES cases(id) ON DELETE CASCADE,

workspace\_id UUID REFERENCES workspaces(id),

category\_id UUID REFERENCES document\_categories(id),

title VARCHAR(255) NOT NULL,

description TEXT,

file\_name VARCHAR(255) NOT NULL,

file\_path VARCHAR(500) NOT NULL, -- includes law\_firm\_id in path for isolation

file\_size BIGINT,

file\_type VARCHAR(100),

mime\_type VARCHAR(100),

version INTEGER DEFAULT 1,

parent\_document\_id UUID REFERENCES documents(id), -- for versioning

is\_confidential BOOLEAN DEFAULT false,

is\_client\_accessible BOOLEAN DEFAULT false,

checksum VARCHAR(255), -- for file integrity

uploaded\_by UUID NOT NULL REFERENCES users(id),

created\_at TIMESTAMP WITH TIME ZONE DEFAULT NOW(),

updated\_at TIMESTAMP WITH TIME ZONE DEFAULT NOW()

);

-- Document access permissions within law firms

CREATE TABLE document\_permissions (

id UUID PRIMARY KEY DEFAULT gen\_random\_uuid(),

document\_id UUID NOT NULL REFERENCES documents(id) ON DELETE CASCADE,

user\_id UUID REFERENCES users(id),

role\_id UUID REFERENCES roles(id),

permission\_type VARCHAR(20) NOT NULL, -- view, download, edit, delete

granted\_by UUID NOT NULL REFERENCES users(id),

granted\_at TIMESTAMP WITH TIME ZONE DEFAULT NOW(),

expires\_at TIMESTAMP WITH TIME ZONE,

is\_active BOOLEAN DEFAULT true

);

-- Document activity log within law firms

CREATE TABLE document\_activity (

id UUID PRIMARY KEY DEFAULT gen\_random\_uuid(),

document\_id UUID NOT NULL REFERENCES documents(id) ON DELETE CASCADE,

user\_id UUID NOT NULL REFERENCES users(id),

action VARCHAR(50) NOT NULL, -- upload, view, download, edit, delete, share

ip\_address INET,

user\_agent TEXT,

created\_at TIMESTAMP WITH TIME ZONE DEFAULT NOW()

);

-- =====================================================

-- 7. CALENDAR AND SCHEDULING (Law Firm Scoped)

-- =====================================================

-- Calendar events within law firms

CREATE TABLE calendar\_events (

id UUID PRIMARY KEY DEFAULT gen\_random\_uuid(),

law\_firm\_id UUID NOT NULL REFERENCES law\_firms(id) ON DELETE CASCADE,

workspace\_id UUID REFERENCES workspaces(id),

case\_id UUID REFERENCES cases(id),

title VARCHAR(255) NOT NULL,

description TEXT,

event\_type VARCHAR(50) NOT NULL, -- court\_date, meeting, deadline, appointment

start\_date\_time TIMESTAMP WITH TIME ZONE NOT NULL,

end\_date\_time TIMESTAMP WITH TIME ZONE NOT NULL,

is\_all\_day BOOLEAN DEFAULT false,

location VARCHAR(255),

is\_recurring BOOLEAN DEFAULT false,

recurrence\_rule JSONB, -- RRULE format for recurring events

court\_room VARCHAR(100),

judge\_name VARCHAR(255),

priority VARCHAR(20) DEFAULT 'medium',

status VARCHAR(20) DEFAULT 'scheduled', -- scheduled, completed, cancelled, postponed

created\_by UUID NOT NULL REFERENCES users(id),

created\_at TIMESTAMP WITH TIME ZONE DEFAULT NOW(),

updated\_at TIMESTAMP WITH TIME ZONE DEFAULT NOW()

);

-- Event attendees within law firms

CREATE TABLE event\_attendees (

id UUID PRIMARY KEY DEFAULT gen\_random\_uuid(),

event\_id UUID NOT NULL REFERENCES calendar\_events(id) ON DELETE CASCADE,

user\_id UUID NOT NULL REFERENCES users(id) ON DELETE CASCADE,

attendance\_type VARCHAR(20) DEFAULT 'required', -- required, optional, organizer

response\_status VARCHAR(20) DEFAULT 'pending', -- pending, accepted, declined, tentative

responded\_at TIMESTAMP WITH TIME ZONE,

notes TEXT,

UNIQUE(event\_id, user\_id)

);

-- =====================================================

-- 8. TASK MANAGEMENT (Law Firm Scoped)

-- =====================================================

-- Tasks table within law firms

CREATE TABLE tasks (

id UUID PRIMARY KEY DEFAULT gen\_random\_uuid(),

law\_firm\_id UUID NOT NULL REFERENCES law\_firms(id) ON DELETE CASCADE,

workspace\_id UUID REFERENCES workspaces(id),

case\_id UUID REFERENCES cases(id),

parent\_task\_id UUID REFERENCES tasks(id), -- for subtasks

title VARCHAR(255) NOT NULL,

description TEXT,

task\_type VARCHAR(50) DEFAULT 'general', -- general, court\_prep, research, filing

status VARCHAR(20) DEFAULT 'pending', -- pending, in\_progress, completed, cancelled

priority VARCHAR(20) DEFAULT 'medium',

assigned\_to UUID REFERENCES users(id),

assigned\_by UUID REFERENCES users(id),

due\_date TIMESTAMP WITH TIME ZONE,

estimated\_hours DECIMAL(5,2),

actual\_hours DECIMAL(5,2),

completion\_percentage INTEGER DEFAULT 0,

completed\_at TIMESTAMP WITH TIME ZONE,

created\_at TIMESTAMP WITH TIME ZONE DEFAULT NOW(),

updated\_at TIMESTAMP WITH TIME ZONE DEFAULT NOW()

);

-- Task comments and updates within law firms

CREATE TABLE task\_comments (

id UUID PRIMARY KEY DEFAULT gen\_random\_uuid(),

task\_id UUID NOT NULL REFERENCES tasks(id) ON DELETE CASCADE,

user\_id UUID NOT NULL REFERENCES users(id),

comment TEXT NOT NULL,

hours\_logged DECIMAL(5,2),

created\_at TIMESTAMP WITH TIME ZONE DEFAULT NOW()

);

-- =====================================================

-- 9. NOTIFICATIONS (Law Firm Scoped)

-- =====================================================

-- Notification templates within law firms

CREATE TABLE notification\_templates (

id UUID PRIMARY KEY DEFAULT gen\_random\_uuid(),

law\_firm\_id UUID REFERENCES law\_firms(id), -- null for system-wide templates

name VARCHAR(100) NOT NULL,

type VARCHAR(50) NOT NULL, -- court\_reminder, task\_due, document\_update, etc.

subject\_template TEXT NOT NULL,

body\_template TEXT NOT NULL,

channels VARCHAR(50)[] DEFAULT ARRAY['email'], -- email, sms, in\_app

is\_system\_template BOOLEAN DEFAULT false,

is\_active BOOLEAN DEFAULT true,

created\_at TIMESTAMP WITH TIME ZONE DEFAULT NOW()

);

-- User notification preferences within law firms

CREATE TABLE user\_notification\_preferences (

id UUID PRIMARY KEY DEFAULT gen\_random\_uuid(),

user\_id UUID NOT NULL REFERENCES users(id) ON DELETE CASCADE,

notification\_type VARCHAR(50) NOT NULL,

email\_enabled BOOLEAN DEFAULT true,

sms\_enabled BOOLEAN DEFAULT false,

in\_app\_enabled BOOLEAN DEFAULT true,

advance\_notice\_hours INTEGER DEFAULT 24, -- how many hours before event to notify

created\_at TIMESTAMP WITH TIME ZONE DEFAULT NOW(),

updated\_at TIMESTAMP WITH TIME ZONE DEFAULT NOW(),

UNIQUE(user\_id, notification\_type)

);

-- Notification queue/log within law firms

CREATE TABLE notifications (

id UUID PRIMARY KEY DEFAULT gen\_random\_uuid(),

law\_firm\_id UUID NOT NULL REFERENCES law\_firms(id) ON DELETE CASCADE,

user\_id UUID NOT NULL REFERENCES users(id) ON DELETE CASCADE,

template\_id UUID REFERENCES notification\_templates(id),

type VARCHAR(50) NOT NULL,

channel VARCHAR(20) NOT NULL, -- email, sms, in\_app

subject VARCHAR(255),

body TEXT,

reference\_type VARCHAR(50), -- case, task, event, document

reference\_id UUID,

status VARCHAR(20) DEFAULT 'pending', -- pending, sent, failed, read

sent\_at TIMESTAMP WITH TIME ZONE,

read\_at TIMESTAMP WITH TIME ZONE,

error\_message TEXT,

metadata JSONB DEFAULT '{}',

created\_at TIMESTAMP WITH TIME ZONE DEFAULT NOW()

);

-- =====================================================

-- 10. AUDIT AND ACTIVITY LOGGING (Law Firm Scoped)

-- =====================================================

-- Law firm-level audit log

CREATE TABLE audit\_logs (

id UUID PRIMARY KEY DEFAULT gen\_random\_uuid(),

law\_firm\_id UUID NOT NULL REFERENCES law\_firms(id) ON DELETE CASCADE,

user\_id UUID REFERENCES users(id),

action VARCHAR(100) NOT NULL,

entity\_type VARCHAR(50) NOT NULL, -- user, case, document, etc.

entity\_id UUID,

old\_values JSONB,

new\_values JSONB,

ip\_address INET,

user\_agent TEXT,

created\_at TIMESTAMP WITH TIME ZONE DEFAULT NOW()

);

-- =====================================================

-- 11. SYSTEM CONFIGURATION (Law Firm Scoped)

-- =====================================================

-- Law firm-specific settings

CREATE TABLE law\_firm\_settings (

id UUID PRIMARY KEY DEFAULT gen\_random\_uuid(),

law\_firm\_id UUID NOT NULL REFERENCES law\_firms(id) ON DELETE CASCADE,

key VARCHAR(100) NOT NULL,

value JSONB NOT NULL,

description TEXT,

created\_at TIMESTAMP WITH TIME ZONE DEFAULT NOW(),

updated\_at TIMESTAMP WITH TIME ZONE DEFAULT NOW(),

UNIQUE(law\_firm\_id, key)

);

-- =====================================================

-- 12. USAGE TRACKING AND ANALYTICS

-- =====================================================

-- Track usage per law firm for billing and analytics

CREATE TABLE usage\_metrics (

id UUID PRIMARY KEY DEFAULT gen\_random\_uuid(),

law\_firm\_id UUID NOT NULL REFERENCES law\_firms(id) ON DELETE CASCADE,

metric\_type VARCHAR(50) NOT NULL, -- storage\_used, api\_calls, active\_users, etc.

metric\_value DECIMAL(15,2) NOT NULL,

recorded\_date DATE NOT NULL,

metadata JSONB DEFAULT '{}',

created\_at TIMESTAMP WITH TIME ZONE DEFAULT NOW(),

UNIQUE(law\_firm\_id, metric\_type, recorded\_date)

);

-- Track feature usage per law firm

CREATE TABLE feature\_usage (

id UUID PRIMARY KEY DEFAULT gen\_random\_uuid(),

law\_firm\_id UUID NOT NULL REFERENCES law\_firms(id) ON DELETE CASCADE,

user\_id UUID REFERENCES users(id),

feature\_name VARCHAR(100) NOT NULL,

usage\_count INTEGER DEFAULT 1,

last\_used\_at TIMESTAMP WITH TIME ZONE DEFAULT NOW(),

date\_used DATE NOT NULL,

created\_at TIMESTAMP WITH TIME ZONE DEFAULT NOW(),

UNIQUE(law\_firm\_id, user\_id, feature\_name, date\_used)

);

-- =====================================================

-- 13. INDEXES FOR PERFORMANCE (Multi-Tenant Aware)

-- =====================================================

-- Platform-level indexes

CREATE INDEX idx\_law\_firms\_slug ON law\_firms(slug);

CREATE INDEX idx\_law\_firms\_subscription\_status ON law\_firms(subscription\_status);

CREATE INDEX idx\_platform\_users\_email ON platform\_users(email);

-- Multi-tenant aware indexes (law\_firm\_id first for partition efficiency)

CREATE INDEX idx\_users\_law\_firm\_email ON users(law\_firm\_id, email);

CREATE INDEX idx\_users\_law\_firm\_user\_type ON users(law\_firm\_id, user\_type);

CREATE INDEX idx\_users\_law\_firm\_active ON users(law\_firm\_id, is\_active);

CREATE INDEX idx\_cases\_law\_firm\_status ON cases(law\_firm\_id, status);

CREATE INDEX idx\_cases\_law\_firm\_case\_number ON cases(law\_firm\_id, case\_number);

CREATE INDEX idx\_cases\_law\_firm\_client ON cases(law\_firm\_id, client\_id);

CREATE INDEX idx\_cases\_law\_firm\_lawyer ON cases(law\_firm\_id, lead\_lawyer\_id);

CREATE INDEX idx\_cases\_law\_firm\_created ON cases(law\_firm\_id, created\_at);

CREATE INDEX idx\_documents\_law\_firm\_case ON documents(law\_firm\_id, case\_id);

CREATE INDEX idx\_documents\_law\_firm\_uploaded\_by ON documents(law\_firm\_id, uploaded\_by);

CREATE INDEX idx\_calendar\_events\_law\_firm\_start ON calendar\_events(law\_firm\_id, start\_date\_time);

CREATE INDEX idx\_calendar\_events\_law\_firm\_type ON calendar\_events(law\_firm\_id, event\_type);

CREATE INDEX idx\_tasks\_law\_firm\_assigned ON tasks(law\_firm\_id, assigned\_to);

CREATE INDEX idx\_tasks\_law\_firm\_status ON tasks(law\_firm\_id, status);

CREATE INDEX idx\_tasks\_law\_firm\_due ON tasks(law\_firm\_id, due\_date);

-- RBAC indexes with tenant awareness

CREATE INDEX idx\_user\_roles\_law\_firm\_user ON user\_roles(user\_id, law\_firm\_id);

CREATE INDEX idx\_roles\_law\_firm\_slug ON roles(law\_firm\_id, slug);

CREATE INDEX idx\_workspace\_members\_law\_firm ON workspace\_members(workspace\_id, user\_id);

-- Audit log indexes

CREATE INDEX idx\_audit\_logs\_law\_firm\_created ON audit\_logs(law\_firm\_id, created\_at);

CREATE INDEX idx\_platform\_audit\_logs\_created ON platform\_audit\_logs(created\_at);

-- Usage tracking indexes

CREATE INDEX idx\_usage\_metrics\_law\_firm\_date ON usage\_metrics(law\_firm\_id, recorded\_date);

CREATE INDEX idx\_feature\_usage\_law\_firm\_date ON feature\_usage(law\_firm\_id, date\_used);

-- =====================================================

-- 14. ROW LEVEL SECURITY (RLS) FOR TENANT ISOLATION

-- =====================================================

-- Enable RLS on all law firm tables

ALTER TABLE users ENABLE ROW LEVEL SECURITY;

ALTER TABLE cases ENABLE ROW LEVEL SECURITY;

ALTER TABLE documents ENABLE ROW LEVEL SECURITY;

ALTER TABLE calendar\_events ENABLE ROW LEVEL SECURITY;

ALTER TABLE tasks ENABLE ROW LEVEL SECURITY;

ALTER TABLE workspaces ENABLE ROW LEVEL SECURITY;

ALTER TABLE roles ENABLE ROW LEVEL SECURITY;

ALTER TABLE audit\_logs ENABLE ROW LEVEL SECURITY;

-- Create RLS policies (examples - implement based on your application's session context)

-- You'll need to set the current law\_firm\_id in your application session

-- Example policy for users table

CREATE POLICY users\_law\_firm\_isolation ON users

USING (law\_firm\_id = current\_setting('app.current\_law\_firm\_id')::UUID);

-- Example policy for cases table

CREATE POLICY cases\_law\_firm\_isolation ON cases

USING (law\_firm\_id = current\_setting('app.current\_law\_firm\_id')::UUID);

-- Similar policies should be created for all law firm-scoped tables

-- =====================================================

-- 15. INITIAL DATA SETUP

-- =====================================================

-- Insert default subscription plans

INSERT INTO subscription\_plans (name, slug, description, price\_monthly, price\_yearly, max\_users, max\_cases, max\_storage\_gb, features) VALUES

('Starter', 'starter', 'Perfect for small law firms', 49.99, 499.99, 5, 50, 10,

'["basic\_case\_management", "document\_storage", "calendar", "client\_portal"]'::jsonb),

('Professional', 'professional', 'For growing law firms', 99.99, 999.99, 15, 200, 50,

'["advanced\_case\_management", "document\_storage", "calendar", "client\_portal", "task\_management", "reporting"]'::jsonb),

('Enterprise', 'enterprise', 'For large law firms', 199.99, 1999.99, 50, 1000, 200,

'["all\_features", "api\_access", "custom\_integrations", "priority\_support"]'::jsonb);

-- Insert default platform permissions

INSERT INTO permissions (name, slug, category, description, is\_platform\_permission) VALUES

-- Platform-level permissions

('Manage Law Firms', 'platform.law\_firms.manage', 'platform', 'Create and manage law firm accounts', true),

('View Analytics', 'platform.analytics.view', 'platform', 'View platform-wide analytics', true),

('Manage Billing', 'platform.billing.manage', 'platform', 'Manage billing and subscriptions', true),

('Support Access', 'platform.support.access', 'platform', 'Access law firm accounts for support', true),

-- Law firm-level permissions

('View Users', 'users.view', 'users', 'View user profiles and information', false),

('Create Users', 'users.create', 'users', 'Create new user accounts', false),

('Edit Users', 'users.edit', 'users', 'Edit user information', false),

('Delete Users', 'users.delete', 'users', 'Delete user accounts', false),

('Manage Roles', 'users.manage\_roles', 'users', 'Assign and manage user roles', false),

('View Cases', 'cases.view', 'cases', 'View case information', false),

('Create Cases', 'cases.create', 'cases', 'Create new cases', false),

('Edit Cases', 'cases.edit', 'cases', 'Edit case information', false),

('Delete Cases', 'cases.delete', 'cases', 'Delete cases', false),

('Assign Cases', 'cases.assign', 'cases', 'Assign cases to team members', false),

('View All Cases', 'cases.view\_all', 'cases', 'View all cases in firm', false),

('View Documents', 'documents.view', 'documents', 'View documents', false),

('Upload Documents', 'documents.upload', 'documents', 'Upload new documents', false),

('Edit Documents', 'documents.edit', 'documents', 'Edit document information', false),

('Delete Documents', 'documents.delete', 'documents', 'Delete documents', false),

('Manage Document Permissions', 'documents.manage\_permissions', 'documents', 'Manage document access', false),

('View Calendar', 'calendar.view', 'calendar', 'View calendar events', false),

('Create Events', 'calendar.create', 'calendar', 'Create calendar events', false),

('Edit Events', 'calendar.edit', 'calendar', 'Edit calendar events', false),

('Delete Events', 'calendar.delete', 'calendar', 'Delete calendar events', false),

('View All Calendars', 'calendar.view\_all', 'calendar', 'View all team calendars', false),

('View Tasks', 'tasks.view', 'tasks', 'View tasks', false),

('Create Tasks', 'tasks.create', 'tasks', 'Create new tasks', false),

('Edit Tasks', 'tasks.edit', 'tasks', 'Edit task information', false),

('Delete Tasks', 'tasks.delete', 'tasks', 'Delete tasks', false),

('Assign Tasks', 'tasks.assign', 'tasks', 'Assign tasks to team members', false),

('Manage Firm Settings', 'admin.firm\_settings', 'admin', 'Manage law firm settings', false),

('Manage Workspaces', 'admin.workspaces', 'admin', 'Manage workspaces', false),

('View Audit Logs', 'admin.audit\_logs', 'admin', 'View firm audit logs', false);

-- Function to create default roles for a law firm

CREATE OR REPLACE FUNCTION create\_default\_law\_firm\_roles(firm\_id UUID)

RETURNS VOID AS $$

BEGIN

-- Law Firm Owner role (full access within firm)

INSERT INTO roles (law\_firm\_id, name, slug, description, level, is\_system\_role, permissions)

VALUES (firm\_id, 'Law Firm Owner', 'owner', 'Full firm access and management', 100, true,

(SELECT json\_agg(slug) FROM permissions WHERE is\_platform\_permission = false)::jsonb);

-- Senior Lawyer role

INSERT INTO roles (law\_firm\_id, name, slug, description, level, is\_system\_role, permissions)

VALUES (firm\_id, 'Senior Lawyer', 'senior\_lawyer', 'Senior lawyer with case management and limited admin', 80, true,

'["cases.view\_all", "cases.create", "cases.edit", "cases.assign", "documents.view", "documents.upload", "documents.edit", "calendar.view\_all", "calendar.create", "calendar.edit", "tasks.view", "tasks.create", "tasks.edit", "tasks.assign", "users.view"]'::jsonb);

-- Junior Lawyer role

INSERT INTO roles (law\_firm\_id, name, slug, description, level, is\_system\_role, permissions)

VALUES (firm\_id, 'Junior Lawyer', 'junior\_lawyer', 'Lawyer with assigned case management', 60, true,

'["cases.view", "cases.edit", "documents.view", "documents.upload", "documents.edit", "calendar.view", "calendar.create", "calendar.edit", "tasks.view", "tasks.create", "tasks.edit"]'::jsonb);

-- Assistant role

INSERT INTO roles (law\_firm\_id, name, slug, description, level, is\_system\_role, permissions)

VALUES (firm\_id, 'Assistant', 'assistant', 'Administrative assistant with document and scheduling access', 40, true,

'["cases.view", "documents.view", "documents.upload", "calendar.view", "calendar.create", "calendar.edit", "tasks.view", "tasks.edit"]'::jsonb);

-- Secretary role

INSERT INTO roles (law\_firm\_id, name, slug, description, level, is\_system\_role, permissions)

VALUES (firm\_id, 'Secretary', 'secretary', 'Secretary with basic administrative access', 30, true,

'["cases.view", "documents.view", "calendar.view", "calendar.create", "tasks.view"]'::jsonb);

-- Client role

INSERT INTO roles (law\_firm\_id, name, slug, description, level, is\_system\_role, permissions)

VALUES (firm\_id, 'Client', 'client', 'Client with limited access to their own cases', 10, true,

'["cases.view", "documents.view", "calendar.view"]'::jsonb);

END;

$$ LANGUAGE plpgsql;

-- Function to initialize a new law firm with default data

CREATE OR REPLACE FUNCTION initialize\_law\_firm(firm\_id UUID)

RETURNS VOID AS $$

BEGIN

-- Create default roles

PERFORM create\_default\_law\_firm\_roles(firm\_id);

-- Create default workspace

INSERT INTO workspaces (law\_firm\_id, name, description, workspace\_type)

VALUES (firm\_id, 'General', 'Default workspace for the law firm', 'general');

-- Create default case categories

INSERT INTO case\_categories (law\_firm\_id, name, description, color) VALUES

(firm\_id, 'Civil Litigation', 'Civil court cases', '#3B82F6'),

(firm\_id, 'Criminal Defense', 'Criminal court cases', '#EF4444'),

(firm\_id, 'Corporate Law', 'Business and corporate matters', '#10B981'),

(firm\_id, 'Family Law', 'Family and domestic matters', '#F59E0B'),

(firm\_id, 'Real Estate', 'Property and real estate matters', '#8B5CF6');

-- Create default document categories

INSERT INTO document\_categories (law\_firm\_id, name, description) VALUES

(firm\_id, 'Pleadings', 'Court filings and pleadings'),

(firm\_id, 'Contracts', 'Contracts and agreements'),

(firm\_id, 'Evidence', 'Evidence and exhibits'),

(firm\_id, 'Correspondence', 'Letters and correspondence'),

(firm\_id, 'Research', 'Legal research and memos');

END;

$$ LANGUAGE plpgsql;

-- =====================================================

-- 16. TRIGGERS FOR AUTOMATIC UPDATES

-- =====================================================

-- Function to update updated\_at timestamp

CREATE OR REPLACE FUNCTION update\_updated\_at\_column()

RETURNS TRIGGER AS $$

BEGIN

NEW.updated\_at = NOW();

RETURN NEW;

END;

$$ LANGUAGE plpgsql;

-- Apply updated\_at trigger to relevant tables

CREATE TRIGGER update\_law\_firms\_updated\_at BEFORE UPDATE ON law\_firms FOR EACH ROW EXECUTE FUNCTION update\_updated\_at\_column();

CREATE TRIGGER update\_workspaces\_updated\_at BEFORE UPDATE ON workspaces FOR EACH ROW EXECUTE FUNCTION update\_updated\_at\_column();

CREATE TRIGGER update\_users\_updated\_at BEFORE UPDATE ON users FOR EACH ROW EXECUTE FUNCTION update\_updated\_at\_column();

CREATE TRIGGER update\_roles\_updated\_at BEFORE UPDATE ON roles FOR EACH ROW EXECUTE FUNCTION update\_updated\_at\_column();

CREATE TRIGGER update\_cases\_updated\_at BEFORE UPDATE ON cases FOR EACH ROW EXECUTE FUNCTION update\_updated\_at\_column();

CREATE TRIGGER update\_documents\_updated\_at BEFORE UPDATE ON documents FOR EACH ROW EXECUTE FUNCTION update\_updated\_at\_column();

CREATE TRIGGER update\_calendar\_events\_updated\_at BEFORE UPDATE ON calendar\_events FOR EACH ROW EXECUTE FUNCTION update\_updated\_at\_column();

CREATE TRIGGER update\_tasks\_updated\_at BEFORE UPDATE ON tasks FOR EACH ROW EXECUTE FUNCTION update\_updated\_at\_column();

-- Function to create audit log entries (law firm level)

CREATE OR REPLACE FUNCTION create\_audit\_log()

RETURNS TRIGGER AS $$

BEGIN

INSERT INTO audit\_logs (

law\_firm\_id,

user\_id,

action,

entity\_type,

entity\_id,

old\_values,

new\_values

) VALUES (

COALESCE(NEW.law\_firm\_id, OLD.law\_firm\_id),

current\_setting('app.current\_user\_id', true)::UUID,

TG\_OP,

TG\_TABLE\_NAME,

COALESCE(NEW.id, OLD.id),

CASE WHEN TG\_OP = 'DELETE' THEN row\_to\_json(OLD) ELSE NULL END,

CASE WHEN TG\_OP IN ('INSERT', 'UPDATE') THEN row\_to\_json(NEW) ELSE NULL END

);

RETURN COALESCE(NEW, OLD);

END;

$$ LANGUAGE plpgsql;

-- Function to create platform audit log entries

CREATE OR REPLACE FUNCTION create\_platform\_audit\_log()

RETURNS TRIGGER AS $$

BEGIN

INSERT INTO platform\_audit\_logs (

platform\_user\_id,

law\_firm\_id,

action,

entity\_type,

entity\_id,

old\_values,

new\_values

) VALUES (

current\_setting('app.current\_platform\_user\_id', true)::UUID,

COALESCE(NEW.id, OLD.id), -- assuming this is law\_firms table

TG\_OP,

TG\_TABLE\_NAME,

COALESCE(NEW.id, OLD.id),

CASE WHEN TG\_OP = 'DELETE' THEN row\_to\_json(OLD) ELSE NULL END,

CASE WHEN TG\_OP IN ('INSERT', 'UPDATE') THEN row\_to\_json(NEW) ELSE NULL END

);

RETURN COALESCE(NEW, OLD);

END;

$$ LANGUAGE plpgsql;

-- Apply audit triggers

CREATE TRIGGER audit\_users AFTER INSERT OR UPDATE OR DELETE ON users FOR EACH ROW EXECUTE FUNCTION create\_audit\_log();

CREATE TRIGGER audit\_cases AFTER INSERT OR UPDATE OR DELETE ON cases FOR EACH ROW EXECUTE FUNCTION create\_audit\_log();

CREATE TRIGGER audit\_documents AFTER INSERT OR UPDATE OR DELETE ON documents FOR EACH ROW EXECUTE FUNCTION create\_audit\_log();

CREATE TRIGGER audit\_user\_roles AFTER INSERT OR UPDATE OR DELETE ON user\_roles FOR EACH ROW EXECUTE FUNCTION create\_audit\_log();

-- Platform-level audit triggers

CREATE TRIGGER audit\_law\_firms AFTER INSERT OR UPDATE OR DELETE ON law\_firms FOR EACH ROW EXECUTE FUNCTION create\_platform\_audit\_log();

-- =====================================================

-- 17. UTILITY FUNCTIONS FOR SAAS MANAGEMENT

-- =====================================================

-- Function to check law firm usage limits

CREATE OR REPLACE FUNCTION check\_law\_firm\_limits(firm\_id UUID, limit\_type VARCHAR(50))

RETURNS BOOLEAN AS $$

DECLARE

current\_count INTEGER;

max\_allowed INTEGER;

BEGIN

SELECT

CASE limit\_type

WHEN 'users' THEN current\_user\_count

WHEN 'cases' THEN current\_case\_count

WHEN 'storage' THEN current\_storage\_gb::INTEGER

END,

CASE limit\_type

WHEN 'users' THEN max\_users

WHEN 'cases' THEN max\_cases

WHEN 'storage' THEN max\_storage\_gb

END

INTO current\_count, max\_allowed

FROM law\_firms

WHERE id = firm\_id;

RETURN current\_count < max\_allowed;

END;

$$ LANGUAGE plpgsql;

-- Function to update law firm usage metrics

CREATE OR REPLACE FUNCTION update\_law\_firm\_usage(firm\_id UUID)

RETURNS VOID AS $$

BEGIN

UPDATE law\_firms SET

current\_user\_count = (SELECT COUNT(\*) FROM users WHERE law\_firm\_id = firm\_id AND is\_active = true),

current\_case\_count = (SELECT COUNT(\*) FROM cases WHERE law\_firm\_id = firm\_id AND status != 'archived'),

current\_storage\_gb = (SELECT COALESCE(SUM(file\_size), 0) / (1024^3) FROM documents WHERE law\_firm\_id = firm\_id),

updated\_at = NOW()

WHERE id = firm\_id;

END;

$$ LANGUAGE plpgsql;