**ReAlign MVP – Database Schema**

**Version:** 1.6

**Date:** May 21, 2025

**Stack:** PostgreSQL + Drizzle ORM (code-first)

**Goal:** Stable, secure, normalized schema with role-scoped access and auditability.

**General Notes:**

* All id fields are UUIDs by default unless specified otherwise, promoting globally unique identifiers.
* Timestamps (created\_at, updated\_at) are used for tracking record creation and modifications.
* CHECK constraints are used to enforce enumerated values for certain fields.
* Foreign key constraints with ON DELETE CASCADE or ON DELETE SET NULL are used where appropriate to maintain relational integrity.

**1. users**

Stores information about all users of the platform.

SQL

CREATE TABLE users (

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

name TEXT,

email TEXT UNIQUE NOT NULL,

phone TEXT, -- For SMS notifications, optional

role TEXT CHECK (role IN ('negotiator', 'seller', 'buyer', 'listing\_agent', 'buyers\_agent', 'escrow')) NOT NULL,

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

updated\_at TIMESTAMP WITH TIME ZONE DEFAULT now()

);

* email: Must be unique.
* role: Defines the user's capabilities within the system.

**2. transactions**

Represents a single short sale transaction.

SQL

CREATE TABLE transactions (

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

title TEXT NOT NULL,

property\_address TEXT NOT NULL, -- As per source [238]

current\_phase TEXT CHECK (current\_phase IN (

'Transaction Initiated', 'Property Listed', 'Initial Document Collection',

'Offer Received', 'Offer Submitted', 'Lender Review',

'BPO Ordered', 'Approval Received', 'In Closing'

)), -- [cite: 238]

created\_by UUID REFERENCES users(id) ON DELETE SET NULL, -- Negotiator who created the transaction

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

updated\_at TIMESTAMP WITH TIME ZONE DEFAULT now()

);

* created\_by: Links to the users table (the negotiator who initiated the transaction).

**3. transaction\_participants**

Links users to transactions, defining their role and status within that specific transaction. This replaces the previous party\_status for clarity on participation.

SQL

CREATE TABLE transaction\_participants (

id SERIAL PRIMARY KEY,

transaction\_id UUID REFERENCES transactions(id) ON DELETE CASCADE NOT NULL,

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

-- Role within this specific transaction, could be a subset of user.role if a user can have multiple roles across system

-- For MVP, this will likely mirror the user's global role if they only participate as that role.

role\_in\_transaction TEXT CHECK (role\_in\_transaction IN ('seller', 'buyer', 'listing\_agent', 'buyers\_agent', 'escrow', 'negotiator\_delegatee')) NOT NULL,

status TEXT CHECK (status IN ('pending', 'complete', 'overdue')), -- Status of their involvement or tasks [cite: 239]

last\_action TEXT, -- Description of the last notable action taken by this participant in this transaction [cite: 239]

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

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

UNIQUE (transaction\_id, user\_id, role\_in\_transaction) -- Ensures a user plays a unique role per transaction

);

* This table maps users to specific transactions they are involved in.
* status and last\_action relate to the participant's overall engagement in the transaction.

**4. messages**

Stores messages exchanged within a transaction.

SQL

CREATE TABLE messages (

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

transaction\_id UUID REFERENCES transactions(id) ON DELETE CASCADE NOT NULL,

sender\_id UUID REFERENCES users(id) ON DELETE SET NULL, -- User who sent the message

text TEXT NOT NULL,

reply\_to UUID REFERENCES messages(id) ON DELETE SET NULL, -- For threaded replies

is\_seed\_message BOOLEAN DEFAULT false, -- True if this is the initial welcome message [cite: 240]

created\_at TIMESTAMP WITH TIME ZONE DEFAULT now()

);

* reply\_to: Self-references for simple threading.
* is\_seed\_message: Indicates if this was the initial message created automatically with the transaction.

**5. document\_requests**

Tracks requests made by negotiators for specific documents.

SQL

CREATE TABLE document\_requests (

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

transaction\_id UUID REFERENCES transactions(id) ON DELETE CASCADE NOT NULL,

requested\_by UUID REFERENCES users(id) ON DELETE SET NULL, -- Negotiator who made the request

assigned\_to UUID REFERENCES users(id) ON DELETE CASCADE, -- User responsible for uploading the document

doc\_type TEXT NOT NULL, -- Type of document requested (from a predefined list for MVP) [cite: 241]

status TEXT CHECK (status IN ('requested', 'pending', 'complete', 'overdue')) NOT NULL, -- Current status of the request [cite: 241]

revision\_note TEXT, -- Note from negotiator if status is reset to 'pending' for revisions

due\_date DATE,

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

updated\_at TIMESTAMP WITH TIME ZONE DEFAULT now()

);

* doc\_type: Sourced from a hardcoded list of templates in MVP.
* **revision\_note**: New field to store negotiator feedback when a document needs revision.
* status: 'pending' can be used when a revision is needed.

**6. uploads**

Stores metadata about uploaded files.

SQL

CREATE TABLE uploads (

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

transaction\_id UUID REFERENCES transactions(id) ON DELETE CASCADE NOT NULL,

uploaded\_by UUID REFERENCES users(id) ON DELETE SET NULL, -- User who uploaded the file

document\_request\_id UUID REFERENCES document\_requests(id) ON DELETE SET NULL, -- Optional: link to a specific request

file\_url TEXT NOT NULL, -- URL to the file in Supabase Storage

file\_name TEXT, -- Original name of the file

content\_type TEXT, -- MIME type of the file

size\_bytes BIGINT, -- File size in bytes

doc\_type TEXT, -- User-defined or selected document type, could mirror a request or be general [cite: 242]

visibility TEXT CHECK (visibility IN ('private', 'shared')) NOT NULL, -- [cite: 242]

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

updated\_at TIMESTAMP WITH TIME ZONE DEFAULT now()

);

* document\_request\_id: Optionally links an upload to a specific document request.
* doc\_type: Can be general or specifically tied to a document\_requests.doc\_type. It's advisable that for uploads fulfilling a request, this doc\_type aligns.
* visibility: Controls who can view the file ('private' typically means uploader and negotiator(s); 'shared' means all transaction participants).

**7. Audit Logging Strategy (MVP)**

For the MVP, detailed audit trails will primarily leverage Supabase's built-in logging capabilities and application-level event logging for critical actions. A dedicated, comprehensive audit\_logs table as described in some security plans might be a post-MVP enhancement if deeper, custom querying is required beyond what Supabase offers.

**Key Events to be Logged (using Supabase logs or application logs):**

* User authentication events (login success/failure, magic link generation/use).
* Transaction creation and phase changes.
* Party invitations and status updates.
* Document request creation, status changes (including revision notes).
* File uploads and visibility changes.
* Notification dispatch events (what was sent, to whom, when).
* Key errors or security-relevant events.

This approach ensures that essential activities are tracked for debugging, security monitoring, and basic audit needs during the MVP phase, aligning with the goal of "Basic event logs + future audit logs table."