Tables (8)

Name	Type	Schema
scdf_annotations		CREATE TABLE scdf_annotations (id INTEGER PRIMARY KEY AUTOINCREMENT, target_type TEXT NOT NULL, -- 'window', 'control', 'event' target_id INTEGER NOT NULL, source TEXT, -- 'ai', 'user', 'scanner' note TEXT, created_at DATETIME DEFAULT CURRENT_TIMESTAMP)
id	INTEGER	"id" INTEGER
target_type	TEXT	"target_type" TEXT NOT NULL
target_id	INTEGER	"target_id" INTEGER NOT NULL
source	TEXT	"source" TEXT
note	TEXT	"note" TEXT
created_at	DATETIME	"created_at" DATETIME DEFAULT CURRENT_TIMESTAMP
scdf_applications		CREATE TABLE scdf_applications (id INTEGER PRIMARY KEY AUTOINCREMENT, name TEXT NOT NULL, version TEXT, platform TEXT, -- Windows, Linux, Web, etc. description TEXT, created_at DATETIME DEFAULT CURRENT_TIMESTAMP)
id	INTEGER	"id" INTEGER
name	TEXT	"name" TEXT NOT NULL
version	TEXT	"version" TEXT
platform	TEXT	"platform" TEXT
description	TEXT	"description" TEXT
created_at	DATETIME	"created_at" DATETIME DEFAULT CURRENT_TIMESTAMP
scdf_change_log		CREATE TABLE scdf_change_log (id INTEGER PRIMARY KEY AUTOINCREMENT, entity_type TEXT NOT NULL, entity_id INTEGER NOT NULL, change_type TEXT, -- add, update, delete changed_by TEXT, -- ai_agent_name, user_id, etc. change_note TEXT, changed_at DATETIME DEFAULT CURRENT_TIMESTAMP)
id	INTEGER	"id" INTEGER
entity_type	TEXT	"entity_type" TEXT NOT NULL
entity_id	INTEGER	"entity_id" INTEGER NOT NULL
change_type	TEXT	"change_type" TEXT
changed_by	TEXT	"changed_by" TEXT
change_note	TEXT	"change_note" TEXT
changed_at	DATETIME	"changed_at" DATETIME DEFAULT CURRENT_TIMESTAMP
scdf_control_events		CREATE TABLE scdf_control_events (id INTEGER PRIMARY KEY AUTOINCREMENT, control_id INTEGER NOT NULL, event_type TEXT NOT NULL, -- click, change, focus, hover, etc. action_description TEXT, ai_generated BOOLEAN DEFAULT 0, FOREIGN KEY (control_id) REFERENCES scdf_controls(id) ON DELETE CASCADE)
id	INTEGER	"id" INTEGER
control_id	INTEGER	"control_id" INTEGER NOT NULL
event_type	TEXT	"event_type" TEXT NOT NULL
action_description	TEXT	"action_description" TEXT
ai_generated	BOOLEAN	"ai_generated" BOOLEAN DEFAULT 0
scdf_controls		CREATE TABLE scdf_controls (id INTEGER PRIMARY KEY AUTOINCREMENT, window_id INTEGER NOT NULL, name TEXT, type TEXT NOT NULL, -- button, textbox, checkbox, grid, etc. label TEXT, x INTEGER, y INTEGER, width INTEGER, height INTEGER, is_enabled BOOLEAN DEFAULT 1, is_visible BOOLEAN DEFAULT 1, data_binding TEXT, -- e.g., bound variable or field FOREIGN KEY (window_id) REFERENCES scdf_windows(id) ON DELETE CASCADE)
id	INTEGER	"id" INTEGER
window_id	INTEGER	"window_id" INTEGER NOT NULL
name	TEXT	"name" TEXT

Name	Type	Schema
type	TEXT	"type" TEXT NOT NULL
label	TEXT	"label" TEXT
x	INTEGER	"x" INTEGER
y	INTEGER	"y" INTEGER
width	INTEGER	"width" INTEGER
height	INTEGER	"height" INTEGER
is_enabled	BOOLEAN	"is_enabled" BOOLEAN DEFAULT 1
is_visible	BOOLEAN	"is_visible" BOOLEAN DEFAULT 1
data_binding	TEXT	"data_binding" TEXT
scdf_window_links		`CREATE TABLE scdf_window_links (id INTEGER PRIMARY KEY AUTOINCREMENT, from_window_id INTEGER NOT NULL, to_window_id INTEGER NOT NULL, trigger_control_id INTEGER, -- optional: which control causes this link condition TEXT, -- e.g., "after login" FOREIGN KEY (from_window_id) REFERENCES scdf_windows(id) ON DELETE CASCADE, FOREIGN KEY (to_window_id) REFERENCES scdf_windows(id) ON DELETE CASCADE, FOREIGN KEY (trigger_control_id) REFERENCES scdf_controls(id))`
id	INTEGER	"id" INTEGER
from_window_id	INTEGER	"from_window_id" INTEGER NOT NULL
to_window_id	INTEGER	"to_window_id" INTEGER NOT NULL
trigger_control_id	INTEGER	"trigger_control_id" INTEGER
condition	TEXT	"condition" TEXT
scdf_windows		`CREATE TABLE scdf_windows ( id INTEGER PRIMARY KEY AUTOINCREMENT, app_id INTEGER NOT NULL, name TEXT, title TEXT, type TEXT, -- main, dialog, modal, etc. description TEXT, UNIQUE(app_id, name), FOREIGN KEY (app_id) REFERENCES scdf_applications(id) ON DELETE CASCADE )`
id	INTEGER	"id" INTEGER
app_id	INTEGER	"app_id" INTEGER NOT NULL
name	TEXT	"name" TEXT
title	TEXT	"title" TEXT
type	TEXT	"type" TEXT
description	TEXT	"description" TEXT
sqlite_sequence		`CREATE TABLE sqlite_sequence(name,seq)`
name		"name"
seq		"seq"

Indices (0)

Name	Type	Schema

Views (0)

Name	Type	Schema

Triggers (0)

Name	Type	Schema