TOY_TRIAL_SETUP_GUIDE

December 18, 2025 at 06:08 AM

Toy Clinical Trial: ZZedc Quick Start Guide

Study: Active vs. Placebo Memory Enhancement Study **Population**: 20 subjects (single lab)
Duration: 3 visits (baseline, week 4, week 8) **Primary Assessment**: MMSE (Mini-Cog Mental State Examination)

This guide walks you through creating and running a complete, minimal clinical trial in ZZedc from scratch.

Part 1: Trial Structure & Planning

Study Overview

```
Timeline:
  Visit 1 (Baseline) → Randomization → Visit 2 (Week 4) → Visit 3 (Week 8)

Forms Needed:
  1. Enrollment Form (baseline visit)
  2. Randomization Form (randomize to Active/Placebo)
  3. MMSE Assessment Form (3 timepoints)

Data to Collect:
  - Demographics (age, gender)
  - Treatment assignment
  - MMSE scores (3 visits)
  - Adverse events (simple yes/no)

Target Sample: 20 subjects
  - 10 Active arm
  - 10 Placebo arm
```

Part 2: Create Study Directory Structure

Step 1: Create Project Directory

```
# Create directory
mkdir -p ~/zzedc_toy_trial
cd ~/zzedc_toy_trial

# Create subdirectories
mkdir -p data forms csv_templates logs exports
```

Step 2: Create Directory Listing

```
ls -la ~/zzedc_toy_trial/
```

Expected output:

```
data/              # Database files
forms/             # Form definitions
csv_templates/     # Data import templates
logs/              # Application logs
exports/           # Export files
```

Part 3: Create Data Dictionary (CSV Format)

Step 3: Create Enrollment Form Definition

File: ~/zzedc_toy_trial/csv_templates/form_enrollment.csv

```
field_name,field_label,field_type,required,validation,description
subject_id,Subject ID,text,TRUE,^S[0-9]{3}$,Unique identifier (format: SXXX)
enrollment_date,Enrollment Date,date,TRUE,NA,Date of enrollment visit
age,Age (years),numeric,TRUE,between 50 and 90,Age at baseline
gender,Gender,select,TRUE,in(M,F),Biological sex
education_years,Years of Education,numeric,FALSE,between 0 and 30,Years of formal education
medical_history,Has Medical History,select,FALSE,in(Yes,No),Any significant medical conditions
```

How to create this file:

```
cat > ~/zzedc_toy_trial/csv_templates/form_enrollment.csv << 'EOF'
field_name,field_label,field_type,required,validation,description
subject_id,Subject ID,text,TRUE,^S[0-9]{3}$,Unique identifier (format: SXXX)
enrollment_date,Enrollment Date,date,TRUE,NA,Date of enrollment visit
age,Age (years),numeric,TRUE,between 50 and 90,Age at baseline
gender,Gender,select,TRUE,in(M,F),Biological sex
education_years,Years of Education,numeric,FALSE,between 0 and 30,Years of formal education
medical_history,Has Medical History,select,FALSE,in(Yes,No),Any significant medical conditions
EOF
```

Step 4: Create Randomization Form Definition

File: ~/zzedc_toy_trial/csv_templates/form_randomization.csv

```
cat > ~/zzedc_toy_trial/csv_templates/form_randomization.csv << 'EOF'
field_name,field_label,field_type,required,validation,description
randomization_date,Randomization Date,date,TRUE,NA,Date subject randomized
treatment_assignment,Treatment Assignment,select,TRUE,in(Active,Placebo),Random assignment to t
study_coordinator,Study Coordinator,text,TRUE,NA,Name of coordinator performing randomization
EOF
```

Step 5: Create MMSE Assessment Form Definition

File: ~/zzedc_toy_trial/csv_templates/form_mmse.csv

```
cat > ~/zzedc_toy_trial/csv_templates/form_mmse.csv << 'EOF'
field_name,field_label,field_type,required,validation,description
visit_label,Visit,select,TRUE,in(Baseline,Week 4,Week 8),Study visit
assessment_date,Assessment Date,date,TRUE,NA,Date of MMSE assessment
orientation_time,Orientation to Time,numeric,TRUE,between 0 and 5,Score (0-5 points)
orientation_place,Orientation to Place,numeric,TRUE,between 0 and 5,Score (0-5 points)
registration,Registration,numeric,TRUE,between 0 and 3,Score (0-3 points)
attention,Attention and Calculation,numeric,TRUE,between 0 and 5,Score (0-5 points)
recall,Recall,numeric,TRUE,between 0 and 3,Score (0-3 points)
language,Language,numeric,TRUE,between 0 and 8,Score (0-8 points)
visual_spatial,Visual-Spatial,numeric,TRUE,between 0 and 1,Score (0-1 point)
mmse_total,MMSE Total Score,numeric,TRUE,between 0 and 30,Sum of all components (calculated)
adverse_events,Adverse Events,select,FALSE,in(Yes,No),Any adverse events reported?
ae_description,AE Description,text,FALSE,NA,Description if adverse events reported
EOF
```

Part 4: Create Test Subject Data

Step 6: Create Subject List CSV

File: ~/zzedc_toy_trial/csv_templates/subjects.csv

Create 20 test subjects with realistic demographic data:

```
cat > ~/zzedc_toy_trial/csv_templates/subjects.csv << 'EOF'
subject_id,enrollment_date,age,gender,education_years,medical_history
S001,2025-12-15,65,M,16,No
S002,2025-12-15,72,F,14,Yes
S003,2025-12-16,58,M,18,No
S004,2025-12-16,68,F,16,Yes
S005,2025-12-17,61,M,12,No
S006,2025-12-17,75,F,14,No
```

```
S007,2025-12-18,70,M,16,Yes
S008,2025-12-18,64,F,18,No
S009,2025-12-19,69,M,14,Yes
S010,2025-12-19,73,F,16,No
S011,2025-12-20,60,M,18,No
S012,2025-12-20,67,F,12,Yes
S013,2025-12-21,62,M,16,No
S014,2025-12-21,76,F,14,Yes
S015,2025-12-22,71,M,18,No
S016,2025-12-22,65,F,16,No
S017,2025-12-23,59,M,14,Yes
S018,2025-12-23,74,F,18,No
S019,2025-12-24,68,M,16,Yes
S020,2025-12-24,70,F,12,No
EOF
```

Step 7: Create Randomization Schedule

File: ~/zzedc_toy_trial/csv_templates/randomization.csv

```
cat > ~/zzedc_toy_trial/csv_templates/randomization.csv << 'EOF'
subject_id,randomization_date,treatment_assignment,study_coordinator
S001,2025-12-15,Active,Jane Smith
S002,2025-12-15,Placebo,Jane Smith
S003,2025-12-16,Active,Bob Johnson
S004,2025-12-16,Placebo,Bob Johnson
S005,2025-12-17,Active,Jane Smith
S006,2025-12-17,Placebo,Jane Smith
S007,2025-12-18,Active,Bob Johnson
S008,2025-12-18,Placebo,Bob Johnson
S009,2025-12-19,Active,Jane Smith
S010,2025-12-19,Placebo,Jane Smith
S011,2025-12-20,Active,Bob Johnson
S012,2025-12-20,Placebo,Bob Johnson
S013,2025-12-21,Active,Jane Smith
S014,2025-12-21,Placebo,Jane Smith
S015,2025-12-22,Active,Bob Johnson
S016,2025-12-22,Placebo,Bob Johnson
S017,2025-12-23,Active,Jane Smith
S018,2025-12-23,Placebo,Jane Smith
S019,2025-12-24,Active,Bob Johnson
S020,2025-12-24,Placebo,Bob Johnson
EOF
```

Step 8: Create MMSE Assessment Data

File: ~/zzedc_toy_trial/csv_templates/mmse_assessments.csv

Create simulated MMSE data showing improvement in Active arm, stability in Placebo:

```
cat > ~/zzedc_toy_trial/csv_templates/mmse_assessments.csv << 'EOF'
subject_id,visit_label,assessment_date,orientation_time,orientation_place,registration,attentio
S001,Baseline,2025-12-15,5,5,3,5,3,8,1,30,No,
S001,Week 4,2026-01-12,5,5,3,5,3,8,1,30,No,
S001,Week 8,2026-02-09,5,5,3,5,3,8,1,30,No,
S002,Baseline,2025-12-15,4,5,3,4,2,7,1,26,No,
S002,Week 4,2026-01-12,4,5,3,4,2,7,1,26,No,
S002,Week 8,2026-02-09,4,5,3,4,2,7,0,25,Yes,Mild headache
S003,Baseline,2025-12-16,5,5,3,5,3,8,1,30,No,
S003,Week 4,2026-01-13,5,5,3,5,3,8,1,30,No,
S003,Week 8,2026-02-10,5,5,3,5,3,8,1,30,No,
S004,Baseline,2025-12-16,4,4,2,4,2,7,1,24,No,
S004,Week 4,2026-01-13,4,4,2,4,2,7,1,24,No,
S004,Week 8,2026-02-10,4,4,2,4,2,7,1,24,No,
S005,Baseline,2025-12-17,5,5,3,5,3,8,1,30,No,
S005,Week 4,2026-01-14,5,5,3,5,3,8,1,30,No,
S005,Week 8,2026-02-11,5,5,3,5,3,8,1,30,No,
S006,Baseline,2025-12-17,3,5,3,3,2,6,1,23,No,
S006,Week 4,2026-01-14,3,5,3,3,2,6,1,23,No,
S006,Week 8,2026-02-11,3,5,3,3,2,6,1,23,No,
S007,Baseline,2025-12-18,5,5,3,4,3,8,1,29,No,
S007,Week 4,2026-01-15,5,5,3,5,3,8,1,30,No,
S007,Week 8,2026-02-12,5,5,3,5,3,8,1,30,No,
S008,Baseline,2025-12-18,4,4,3,4,2,7,1,25,No,
S008,Week 4,2026-01-15,4,4,3,4,2,7,1,25,No,
S008,Week 8,2026-02-12,4,4,3,4,2,7,1,25,No,
S009,Baseline,2025-12-19,5,5,3,5,3,8,1,30,No,
S009,Week 4,2026-01-16,5,5,3,5,3,8,1,30,No,
S009,Week 8,2026-02-13,5,5,3,5,3,8,1,30,No,
S010,Baseline,2025-12-19,3,4,2,3,1,6,1,20,No,
S010,Week 4,2026-01-16,3,4,2,3,1,6,1,20,No,
S010,Week 8,2026-02-13,3,4,2,3,1,6,1,20,No,
S011,Baseline,2025-12-20,5,5,3,5,3,8,1,30,No,
S011,Week 4,2026-01-17,5,5,3,5,3,8,1,30,No,
S011,Week 8,2026-02-14,5,5,3,5,3,8,1,30,No,
S012,Baseline,2025-12-20,4,5,3,4,2,7,1,26,No,
S012,Week 4,2026-01-17,4,5,3,4,2,7,1,26,No,
S012,Week 8,2026-02-14,4,5,3,4,2,7,1,26,No,
S013,Baseline,2025-12-21,5,5,3,4,3,8,1,29,No,
S013,Week 4,2026-01-18,5,5,3,5,3,8,1,30,No,
S013,Week 8,2026-02-15,5,5,3,5,3,8,1,30,No,
```

```
S014,Baseline,2025-12-21,4,4,2,4,2,7,1,24,No,
S014,Week 4,2026-01-18,4,4,2,4,2,7,1,24,No,
S014,Week 8,2026-02-15,4,4,2,4,2,7,0,23,No,
S015,Baseline,2025-12-22,5,5,3,5,3,8,1,30,No,
S015,Week 4,2026-01-19,5,5,3,5,3,8,1,30,No,
S015,Week 8,2026-02-16,5,5,3,5,3,8,1,30,No,
S016,Baseline,2025-12-22,3,5,3,3,2,6,1,23,No,
S016,Week 4,2026-01-19,3,5,3,3,2,6,1,23,No,
S016,Week 8,2026-02-16,3,5,3,3,2,6,1,23,No,
S017,Baseline,2025-12-23,5,5,3,5,3,8,1,30,No,
S017,Week 4,2026-01-20,5,5,3,5,3,8,1,30,Yes,Dizziness
S017,Week 8,2026-02-17,5,5,3,5,3,8,1,30,No,
S018,Baseline,2025-12-23,4,4,3,4,2,7,1,25,No,
S018,Week 4,2026-01-20,4,4,3,4,2,7,1,25,No,
S018,Week 8,2026-02-17,4,4,3,4,2,7,1,25,No,
S019,Baseline,2025-12-24,5,5,3,4,3,8,1,29,No,
S019,Week 4,2026-01-21,5,5,3,5,3,8,1,30,No,
S019,Week 8,2026-02-18,5,5,3,5,3,8,1,30,No,
S020,Baseline,2025-12-24,3,4,2,3,1,6,1,20,No,
S020,Week 4,2026-01-21,3,4,2,3,1,6,1,20,No,
S020,Week 8,2026-02-18,3,4,2,3,1,6,1,20,No,
EOF
```

Part 5: Setup ZZedc Database

Step 9: Create R Setup Script

File: ~/zzedc_toy_trial/setup_toy_trial.R

```
cat > ~/zzedc_toy_trial/setup_toy_trial.R << 'EOF'
#!/usr/bin/env Rscript

# Toy Clinical Trial Setup Script
# Creates complete ZZedc database with 20 subjects, randomization, and MMSE data

cat("Creating Toy Clinical Trial Database...\n")
cat("=======================================\n\n")

# Load required packages
library(DBI)
library(RSQLite)
library(digest)

# Paths
db_path <- "./data/toy_trial.db"
```

```r
csv_dir <- "./csv_templates"

# Create data directory if needed
if (!dir.exists("data")) dir.create("data", recursive = TRUE)

# 1. CREATE DATABASE
cat("Step 1: Creating SQLite database...\n")
conn <- dbConnect(SQLite(), db_path)

# 2. CREATE STUDY_INFO TABLE
cat("Step 2: Creating study information table...\n")
dbExecute(conn, "
  CREATE TABLE IF NOT EXISTS study_info (
    study_id TEXT PRIMARY KEY,
    study_name TEXT NOT NULL,
    protocol_id TEXT UNIQUE NOT NULL,
    principal_investigator TEXT,
    pi_email TEXT,
    study_phase TEXT,
    target_enrollment INTEGER,
    created_date TIMESTAMP DEFAULT CURRENT_TIMESTAMP
  )
")

# Insert study info
study_id <- "TOY-TRIAL-001"
dbExecute(conn, "
  INSERT OR REPLACE INTO study_info
  (study_id, study_name, protocol_id, principal_investigator, pi_email,
   study_phase, target_enrollment)
  VALUES (?, ?, ?, ?, ?, ?, ?)
", list(
  study_id,
  "Active vs. Placebo Memory Enhancement Study",
  "PROTO-2025-001",
  "Dr. John Smith",
  "john.smith@hospital.org",
  "Phase III",
  20
))

cat("  Study: TOY-TRIAL-001 created\n")

# 3. CREATE SUBJECTS TABLE
cat("Step 3: Creating subjects table...\n")
```

```r
dbExecute(conn, "
  CREATE TABLE IF NOT EXISTS subjects (
    subject_id TEXT PRIMARY KEY,
    study_id TEXT NOT NULL REFERENCES study_info(study_id),
    enrollment_date TIMESTAMP,
    enrollment_age INTEGER,
    gender TEXT CHECK(gender IN ('M', 'F')),
    status TEXT CHECK(status IN ('Active', 'Completed', 'Withdrawn')),
    created_date TIMESTAMP DEFAULT CURRENT_TIMESTAMP
  )
")

# Read subjects CSV and insert
subjects_csv <- read.csv(file.path(csv_dir, "subjects.csv"))
cat("  Loading", nrow(subjects_csv), "subjects...\n")

for (i in 1:nrow(subjects_csv)) {
  row <- subjects_csv[i, ]
  dbExecute(conn, "
    INSERT INTO subjects
    (subject_id, study_id, enrollment_date, enrollment_age, gender, status)
    VALUES (?, ?, ?, ?, ?, 'Active')
  ", list(
    row$subject_id,
    study_id,
    row$enrollment_date,
    row$age,
    row$gender
  ))
}

# 4. CREATE RANDOMIZATION TABLE
cat("Step 4: Creating randomization table...\n")
dbExecute(conn, "
  CREATE TABLE IF NOT EXISTS randomization (
    randomization_id TEXT PRIMARY KEY,
    subject_id TEXT NOT NULL REFERENCES subjects(subject_id),
    randomization_date TIMESTAMP,
    treatment_assignment TEXT CHECK(treatment_assignment IN ('Active', 'Placebo')),
    study_coordinator TEXT,
    created_date TIMESTAMP DEFAULT CURRENT_TIMESTAMP
  )
")

# Read randomization CSV and insert
```

```r
rand_csv <- read.csv(file.path(csv_dir, "randomization.csv"))
cat("  Loading randomization data for", nrow(rand_csv), "subjects...\n")

for (i in 1:nrow(rand_csv)) {
  row <- rand_csv[i, ]
  rand_id <- paste0("RAND-", row$subject_id, "-", format(Sys.time(), "%Y%m%d"))
  dbExecute(conn, "
    INSERT INTO randomization
    (randomization_id, subject_id, randomization_date, treatment_assignment, study_coordinator)
    VALUES (?, ?, ?, ?, ?)
  ", list(
    rand_id,
    row$subject_id,
    row$randomization_date,
    row$treatment_assignment,
    row$study_coordinator
  ))
}

# 5. CREATE MMSE ASSESSMENTS TABLE
cat("Step 5: Creating MMSE assessments table...\n")
dbExecute(conn, "
  CREATE TABLE IF NOT EXISTS mmse_assessments (
    assessment_id TEXT PRIMARY KEY,
    subject_id TEXT NOT NULL REFERENCES subjects(subject_id),
    visit_label TEXT CHECK(visit_label IN ('Baseline', 'Week 4', 'Week 8')),
    assessment_date TIMESTAMP,
    orientation_time INTEGER CHECK(orientation_time BETWEEN 0 AND 5),
    orientation_place INTEGER CHECK(orientation_place BETWEEN 0 AND 5),
    registration INTEGER CHECK(registration BETWEEN 0 AND 3),
    attention INTEGER CHECK(attention BETWEEN 0 AND 5),
    recall INTEGER CHECK(recall BETWEEN 0 AND 3),
    language INTEGER CHECK(language BETWEEN 0 AND 8),
    visual_spatial INTEGER CHECK(visual_spatial BETWEEN 0 AND 1),
    mmse_total INTEGER CHECK(mmse_total BETWEEN 0 AND 30),
    adverse_events TEXT CHECK(adverse_events IN ('Yes', 'No')),
    ae_description TEXT,
    created_date TIMESTAMP DEFAULT CURRENT_TIMESTAMP
  )
")

# Read MMSE CSV and insert
mmse_csv <- read.csv(file.path(csv_dir, "mmse_assessments.csv"))
cat("  Loading", nrow(mmse_csv), "MMSE assessments...\n")
```

```r
for (i in 1:nrow(mmse_csv)) {
  row <- mmse_csv[i, ]
  assessment_id <- paste0("MMSE-", row$subject_id, "-",
                          format(as.Date(row$assessment_date), "%Y%m%d"))
  dbExecute(conn, "
    INSERT INTO mmse_assessments
    (assessment_id, subject_id, visit_label, assessment_date,
     orientation_time, orientation_place, registration, attention,
     recall, language, visual_spatial, mmse_total, adverse_events, ae_description)
    VALUES (?, ?, ?, ?, ?, ?, ?, ?, ?, ?, ?, ?, ?, ?)
  ", list(
    assessment_id,
    row$subject_id,
    row$visit_label,
    row$assessment_date,
    row$orientation_time,
    row$orientation_place,
    row$registration,
    row$attention,
    row$recall,
    row$language,
    row$visual_spatial,
    row$mmse_total,
    row$adverse_events,
    ifelse(is.na(row$ae_description), "", row$ae_description)
  ))
}

# 6. CREATE INDEXES FOR PERFORMANCE
cat("Step 6: Creating database indexes...\n")
dbExecute(conn, "CREATE INDEX idx_subjects_study ON subjects(study_id)")
dbExecute(conn, "CREATE INDEX idx_rand_subject ON randomization(subject_id)")
dbExecute(conn, "CREATE INDEX idx_mmse_subject ON mmse_assessments(subject_id)")
dbExecute(conn, "CREATE INDEX idx_mmse_visit ON mmse_assessments(visit_label)")
dbExecute(conn, "CREATE INDEX idx_mmse_date ON mmse_assessments(assessment_date)")

# 7. VERIFY DATA
cat("\nStep 7: Verifying data...\n")

subject_count <- dbGetQuery(conn, "SELECT COUNT(*) as count FROM subjects")[1,1]
rand_count <- dbGetQuery(conn, "SELECT COUNT(*) as count FROM randomization")[1,1]
mmse_count <- dbGetQuery(conn, "SELECT COUNT(*) as count FROM mmse_assessments")[1,1]

cat("  Total subjects enrolled:", subject_count, "\n")
cat("  Total randomizations:", rand_count, "\n")
```

```r
cat("  Total MMSE assessments:", mmse_count, "\n")

# Show treatment breakdown
treatment_breakdown <- dbGetQuery(conn, "
  SELECT treatment_assignment, COUNT(*) as count
  FROM randomization
  GROUP BY treatment_assignment
")
cat("\n  Treatment breakdown:\n")
for (i in 1:nrow(treatment_breakdown)) {
  cat("    ", treatment_breakdown[i,1], ":", treatment_breakdown[i,2], "\n")
}

# Show visit breakdown
visit_breakdown <- dbGetQuery(conn, "
  SELECT visit_label, COUNT(*) as count
  FROM mmse_assessments
  GROUP BY visit_label
")
cat("\n  Visit breakdown:\n")
for (i in 1:nrow(visit_breakdown)) {
  cat("    ", visit_breakdown[i,1], ":", visit_breakdown[i,2], "\n")
}

# Close connection
dbDisconnect(conn)

cat("\n=======================================\n")
cat("SUCCESS: Toy trial database created!\n")
cat("Database location:", db_path, "\n")
cat("Ready to launch ZZedc with toy trial data\n")
cat("=======================================\n")
EOF

chmod +x ~/zzedc_toy_trial/setup_toy_trial.R
```

Step 10: Run Setup Script

```
cd ~/zzedc_toy_trial
Rscript setup_toy_trial.R
```

Expected output:

```
Creating Toy Clinical Trial Database...
=======================================

Step 1: Creating SQLite database...
```

```
Step 2: Creating study information table...
   Study: TOY-TRIAL-001 created
Step 3: Creating subjects table...
   Loading 20 subjects...
Step 4: Creating randomization table...
   Loading randomization data for 20 subjects...
Step 5: Creating MMSE assessments table...
   Loading 60 MMSE assessments...
Step 6: Creating database indexes...

Step 7: Verifying data...
   Total subjects enrolled: 20
   Total randomizations: 20
   Total MMSE assessments: 60

   Treatment breakdown:
      Active : 10
      Placebo : 10

   Visit breakdown:
      Baseline : 20
      Week 4 : 20
      Week 8 : 20


========================================
SUCCESS: Toy trial database created!
Database location: ./data/toy_trial.db
Ready to launch ZZedc with toy trial data
========================================
```

Part 6: Create Configuration Files

Step 11: Create Config File

File: ~/zzedc_toy_trial/config.yml

```
cat > ~/zzedc_toy_trial/config.yml << 'EOF'
# ZZedc Configuration for Toy Clinical Trial

database:
  type: sqlite
  path: ./data/toy_trial.db
  pool_size: 5

auth:
```

```
  session_timeout_minutes: 60
  max_failed_attempts: 3

security:
  enforce_https: false
  password_min_length: 6

ui:
  theme: bootstrap
  bootstrap_version: 5
  brand_name: "Toy Trial - ZZedc"

study:
  name: "Active vs. Placebo Memory Enhancement Study"
  protocol_id: "PROTO-2025-001"
  pi_name: "Dr. John Smith"
  target_enrollment: 20

compliance:
  gdpr_enabled: true
  cfr_part11_enabled: false
  enable_audit_logging: true
EOF
```

Part 7: Create Users

Step 12: Create R Script to Add Users

File: ~/zzedc_toy_trial/add_users.R

```
cat > ~/zzedc_toy_trial/add_users.R << 'EOF'
#!/usr/bin/env Rscript

# Add test users to toy trial database

library(DBI)
library(RSQLite)
library(digest)

db_path <- "./data/toy_trial.db"
conn <- dbConnect(SQLite(), db_path)

# Create users table
dbExecute(conn, "
  CREATE TABLE IF NOT EXISTS users (
```

```r
    user_id TEXT PRIMARY KEY,
    username TEXT UNIQUE NOT NULL,
    password_hash TEXT NOT NULL,
    full_name TEXT,
    email TEXT,
    role TEXT,
    active BOOLEAN DEFAULT 1,
    created_date TIMESTAMP DEFAULT CURRENT_TIMESTAMP
  )
")

# Define salt for password hashing
salt <- "toy_trial_salt_2025"

# Add users
users <- data.frame(
  username = c("admin", "jane_smith", "bob_johnson", "researcher"),
  password = c("admin123", "jane123", "bob123", "research123"),
  full_name = c("Admin User", "Jane Smith", "Bob Johnson", "Lead Researcher"),
  email = c("admin@trial.org", "jane@trial.org", "bob@trial.org", "lead@trial.org"),
  role = c("Admin", "Coordinator", "Coordinator", "Researcher"),
  stringsAsFactors = FALSE
)

for (i in 1:nrow(users)) {
  user_id <- paste0("USER-", i)
  pwd_hash <- digest(paste0(users[i, "password"], salt), algo = "sha256")

  dbExecute(conn, "
    INSERT OR REPLACE INTO users
    (user_id, username, password_hash, full_name, email, role, active)
    VALUES (?, ?, ?, ?, ?, ?, 1)
  ", list(
    user_id,
    users[i, "username"],
    pwd_hash,
    users[i, "full_name"],
    users[i, "email"],
    users[i, "role"]
  ))

  cat("Added user:", users[i, "username"], "password:", users[i, "password"], "\n")
}

dbDisconnect(conn)
```

```r
cat("\nUsers created successfully!\n")
cat("Test Credentials:\n")
cat("  admin / admin123\n")
cat("  jane_smith / jane123\n")
cat("  bob_johnson / bob123\n")
cat("  researcher / research123\n")
EOF

chmod +x ~/zzedc_toy_trial/add_users.R
```

Step 13: Run Add Users Script

```r
cd ~/zzedc_toy_trial
Rscript add_users.R
```

Part 8: Create Launch Script

Step 14: Create Launch Script

File: ~/zzedc_toy_trial/launch_toy_trial.R

```r
cat > ~/zzedc_toy_trial/launch_toy_trial.R << 'EOF'
#!/usr/bin/env Rscript

# Launch ZZedc with Toy Trial Database

cat("Launching ZZedc - Toy Clinical Trial\n")
cat("====================================\n\n")

# Set environment variables
Sys.setenv(
  ZZEDC_DB_PATH = "./data/toy_trial.db",
  ZZEDC_CONFIG = "./config.yml",
  ZZEDC_STUDY = "TOY-TRIAL-001"
)

# Load and launch ZZedc
library(zzedc)

cat("Starting ZZedc application...\n")
cat("Database: ./data/toy_trial.db\n")
cat("Configuration: ./config.yml\n")
cat("Opening in browser: http://localhost:3838\n\n")
```

```
# Launch the app
launch_zzedc(
  host = "127.0.0.1",
  port = 3838,
  launch.browser = TRUE
)
EOF

chmod +x ~/zzedc_toy_trial/launch_toy_trial.R
```

Part 9: Launch & Use

Step 15: Launch the Application

```
cd ~/zzedc_toy_trial
Rscript launch_toy_trial.R
```

Step 16: Login and Explore

When the app opens:

1. **Login**: Use credentials from Step 13
   - Username: jane_smith
   - Password: jane123
2. **Navigation**:
   - **Home Tab**: Overview of study
   - **Data Explorer**: Browse 20 subjects
   - **Reports**: View randomization breakdown and MMSE trends
   - **Exports**: Export data in CSV/Excel format
3. **View Data**:
   - See all 20 enrolled subjects
   - View randomization: 10 Active, 10 Placebo
   - Inspect MMSE scores across 3 visits
   - Check adverse events (2 subjects had minor AEs)

Part 10: Verify Trial Structure

Step 17: Query Database to Verify

```
cat > ~/zzedc_toy_trial/verify_toy_trial.R << 'EOF'
#!/usr/bin/env Rscript

# Verify toy trial data
```

```r
library(DBI)
library(RSQLite)

db_path <- "./data/toy_trial.db"
conn <- dbConnect(SQLite(), db_path)

cat("\n=== TOY TRIAL VERIFICATION ===\n\n")

# Study info
cat("Study Information:\n")
study_info <- dbGetQuery(conn, "SELECT * FROM study_info")
print(study_info[, c("study_id", "study_name", "protocol_id", "target_enrollment")])

cat("\n\nSubject Summary:\n")
subjects <- dbGetQuery(conn, "
  SELECT
    COUNT(*) as total_subjects,
    COUNT(DISTINCT gender) as genders,
    ROUND(AVG(enrollment_age), 1) as mean_age,
    MIN(enrollment_age) as min_age,
    MAX(enrollment_age) as max_age
  FROM subjects
")
print(subjects)

cat("\n\nTreatment Assignment:\n")
treatment <- dbGetQuery(conn, "
  SELECT treatment_assignment, COUNT(*) as count
  FROM randomization
  GROUP BY treatment_assignment
")
print(treatment)

cat("\n\nMMSE Scores by Visit and Treatment:\n")
mmse_summary <- dbGetQuery(conn, "
  SELECT
    r.treatment_assignment as treatment,
    m.visit_label as visit,
    COUNT(*) as n_subjects,
    ROUND(AVG(m.mmse_total), 1) as mean_score,
    ROUND(MIN(m.mmse_total), 0) as min_score,
    ROUND(MAX(m.mmse_total), 0) as max_score
  FROM mmse_assessments m
  JOIN randomization r ON m.subject_id = r.subject_id
  GROUP BY r.treatment_assignment, m.visit_label
```

```
    ORDER BY r.treatment_assignment,
             CASE m.visit_label
               WHEN 'Baseline' THEN 1
               WHEN 'Week 4' THEN 2
               WHEN 'Week 8' THEN 3
             END
")
print(mmse_summary)

cat("\n\nAdverse Events:\n")
adverse <- dbGetQuery(conn, "
  SELECT
    m.subject_id,
    r.treatment_assignment,
    m.visit_label,
    m.ae_description
  FROM mmse_assessments m
  JOIN randomization r ON m.subject_id = r.subject_id
  WHERE m.adverse_events = 'Yes' AND m.ae_description != ''
")
if (nrow(adverse) > 0) {
  print(adverse)
} else {
  cat("No adverse events recorded\n")
}

dbDisconnect(conn)

cat("\n=== VERIFICATION COMPLETE ===\n\n")
EOF

chmod +x ~/zzedc_toy_trial/verify_toy_trial.R
Rscript ~/zzedc_toy_trial/verify_toy_trial.R
```

Summary: Complete Directory Structure

After completing all steps, your directory should look like:

```
~/zzedc_toy_trial/
├── data/
│   └── toy_trial.db            # SQLite database (20 subjects, 60 assessments)
├── csv_templates/
│   ├── form_enrollment.csv     # Enrollment form definition
│   ├── form_randomization.csv  # Randomization form definition
```

```
│    ├── form_mmse.csv              # MMSE assessment form definition
│    ├── subjects.csv               # 20 enrolled subjects
│    ├── randomization.csv          # Randomization assignments (10/10 split)
│    └── mmse_assessments.csv       # 60 MMSE assessments (3 visits × 20 subjects)
├── logs/                           # Application logs (created on first run)
├── exports/                        # Export files (created on export)
├── config.yml                      # ZZedc configuration
├── setup_toy_trial.R               # Database setup script
├── add_users.R                     # User creation script
├── launch_toy_trial.R              # Application launcher
└── verify_toy_trial.R              # Data verification script
```

Quick Reference: Commands

Setup (one-time):

```
cd ~/zzedc_toy_trial
Rscript setup_toy_trial.R
Rscript add_users.R
Rscript verify_toy_trial.R
```

Launch (every time):

```
cd ~/zzedc_toy_trial
Rscript launch_toy_trial.R
# Then open http://localhost:3838
```

Verify Data:

```
cd ~/zzedc_toy_trial
Rscript verify_toy_trial.R
```

Next Steps

After setting up the toy trial:

1. **Explore Features**:
   - View data in data explorer
   - Generate reports by treatment arm
   - Export data in multiple formats
2. **Test Encryption** (Feature #1):
   - Replace toy_trial.db with encrypted version
   - Verify data still accessible with correct key
   - Test key rotation

3. **Add More Data**:
   - Enroll additional subjects
   - Add more visits
   - Collect additional assessments
4. **Customize for Real Study**:
   - Modify forms in CSV templates
   - Adjust validation rules
   - Add study-specific fields

Created: December 2025 **Purpose**: Minimal toy example for ZZedc feature demonstration
Sample Size: 20 subjects, 3 visits, 60 total assessments