The Countdown of Conte Carlos – Full Game Infographic

GAME OVERVIEW
• Pygame Zero game where Carlos must match a target histogram or die.
• Two levers: choose a base distribution + choose a function (f(x)).
• Stakes: Timer = Queen's life; wrong answers = fingers lost.

GAME LOGIC (From Code)
• MAX_FINGERS = 6, ROUND_TIME = 60s.
• Distributions (IDs 1–6):
  - Normal, Skewed Right, Skewed Left, Uniform,
    Bimodal Symmetric, Bimodal Asymmetric.
• Functions (IDs 1–4):
  - f(x)=x, x^2, √x, 1−x.
• Answers list maps a target image → (distribution ID, function ID).

PUZZLE SYSTEM
• generate_new_puzzle():
  - Randomly selects 1 of 10 answer combinations.
  - Creates target histogram using scaled Actor() sprite.
  - Resets timer, fingers, lever positions.

USER INPUT (From Code)
• A/D → cycle distributions.
• J/L → cycle functions.
• ENTER → submit guess.
• R → new puzzle.

GAME STATES
• PLAYING, WON, LOST_TIME, LOST_FINGERS.
• Winning: guess matches answer_key → WON
• Losing:
  - Time hits 0 → LOST_TIME.
  - Fingers reach 0 → LOST_FINGERS.

UI ELEMENTS DRAWN
• Target histogram at top.
• Lever 1: Current distribution preview.
• Lever 2: Current function preview.
• Hourglass bar timer (yellow fill).
• Finger counter.
• End-game overlays for WIN / TIME LOSS / FINGER LOSS.

GRAPHICS SYSTEM
• make_scaled_actor(): rescales images to avoid distortion.
• draw(): renders UI, histograms, prompts, and overlays.

HISTOGRAM SYSTEM
• Target histograms are pre-generated images based on valid (distribution, function) pairs.
• Images are scaled using make_scaled_actor() to ensure clean resolution on screen.
• Player previews update in real-time as levers change.

DISTRIBUTION & FUNCTION DESIGN
• All distributions chosen to reflect distinct statistical shapes: symmetric, skewed, multi-modal.
• Functions modify shape dynamically to teach intuitive transformation effects.
• IDs map directly to answer validation logic.

TIMER & FINGER MECHANICS
• 60-second countdown builds urgency.
• Finger system adds tension and clear penalty feedback.
• Messages guide the player through success/failure states.

USER INTERFACE LAYOUT
• Top: Target histogram (central visual anchor)
• Middle left: Lever 1 distribution preview
• Middle right: Lever 2 function preview
• Bottom: Timer bar, finger counter, controls reference
• End screens use color-coded overlays for clarity.

TECHNICAL ARCHITECTURE
• draw() handles all rendering each frame.
• update(dt) manages the timer and win/loss transitions.
• on_key_down() handles all input logic for levers and submitting guesses.
• generate_new_puzzle() resets and initializes new puzzles cleanly.