

Feature Implementation Instructions: Breath Kasina

Overview:

We're adding a new category called "Breath Kasinas" to our existing KASINA application (www.kasina.app). This new category will complement our current Color, Elemental, & Vajrayana Kasina sets, while being distinct among these other sets, in that it's feature we want to any Kasina once it's active. We will begin with the "Expand & Contract" kasina visualization, driven by real-time breath data from the Vernier Go Direct Respiration Belt. Later we'll add "Brighten & Darken" and "Changing Colors" as some other breath kasina effects that you could layer on, along with, or instead of, Expand & Contract.

Hardware Integration

- **Device:** Vernier Go Direct Respiration Belt
- **Communication:** Web Bluetooth API
- **Data Type:** Continuous numeric breath amplitude data
- **Data Update Rate:** ~20 Hz (updates every ~50 ms)

Integration Steps:

1. Connect via Web Bluetooth, using JavaScript in our Replit environment.
 2. Identify BLE Service and Characteristic UUIDs (using nRF Connect or Vernier documentation).
 3. Stream real-time breath amplitude data to the web app.
-

Visualization Details: Breath Kasina: Expand & Contract

- **Primary Visual Element:** A spherical orb. This effect applies to any of the other kasinas built into the app from the other sets.
- **Animation Behavior:**

- Orb expands smoothly with inhalation and contracts gently with exhalation.
- Orb's size directly corresponds to real-time breath amplitude (normalized).
- **Breathing Rate Adaptation:**
 - Orb visual intensity and maximum size dynamically adjust based on user's breathing rate.
 - As the breathing rate decreases, indicative of deeper meditation, the overall visual intensity and orb movements become subtler.
 - Specifically, a breathing rate of ~4 breaths per minute corresponds to near visual stillness, promoting deep concentration and calm.
- **Algorithm for Adaptive Visual Scaling:**

```
function updateOrbVisualization(rawBreathValue, currentBreathsPerMinute) {
  const normalizedAmplitude = (rawBreathValue - minVal) / (maxVal - minVal);

  // Breathing rate-based intensity scaling
  const intensityScale = Math.min(currentBreathsPerMinute / 12, 1); // Normal breath rate
  ~12/min
  const scaleFactor = 1 + (0.3 * normalizedAmplitude * intensityScale);

  orb.scale.setScalar(scaleFactor);

  // At 4 breaths per minute (or lower), intensityScale approaches zero, causing subtle, minimal
  visual motion.
}
```

User Experience (UX)

- **New Category Placement:** Listed alongside "Color Kasinas" and "Elemental Kasinas" in the main selection menu.
- **User Setup:**
 - Button labeled "Connect Breath Sensor" initiates Web Bluetooth connection.

- Short calibration phase (10–20 seconds) to establish personalized min/max breathing amplitude.

Technical Specifications for Replit Implementation

- Use Replit's built-in Node.js and frontend frameworks (Three.js or Canvas 2D).
- Web Bluetooth API for BLE integration: Chrome desktop and Android-compatible.
- Ensure graceful handling of connection errors and provide clear feedback to the user.

Testing & Validation Criteria

- Successfully streaming and visualizing real-time breath data.
- Smooth, responsive visual scaling without noticeable latency (below 100 ms).
- Proper adaptive intensity response verified through breathing rate changes (from normal ~12 breaths/min down to 4 breaths/min).

This implementation is foundational, and once validated, additional kasina visualizations (e.g., Brighten & Darken, Changing Colors, etc.) and further biofeedback integrations (e.g., HRV data via Polar H10) can be considered.